บทที่ 4

ผลการวิจัย

เพื่อเปรียบเทียบประสิทธิภาพการทำงานของอัลกอริทึม LCC จึงทำการทดสอบฝึกสอนโครงข่ายด้วยอัลกอริทึม LCC เปรียบเทียบกับอัลกอริทึม ดังต่อไปนี้

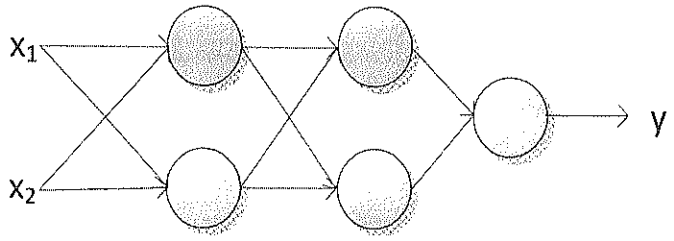
1. LM (Levenberg-Marquartd Backpropagation)
2. GDA (Gradient descent with adaptive learning rule backpropagation)
3. GD (Gradient descent backpropagation)

อัลกอริทึม LM, GDA และ GD นั้น เป็นเครื่องมือ (Toolbox) ในโปรแกรม MATLAB โดยในการทดสอบจะทำการฝึกสอนโครงข่ายจากทั้ง 4 อัลกอริทึม โดยในแต่ละอัลกอริทึมที่ทดสอบกำหนดให้มีสถาปัตยกรรมที่เหมือนกัน โดยที่แต่ละสถาปัตยกรรมของแต่ละอัลกอริทึมจะถูกฝึกสอนจำนวนทั้งหมด 10 รอบ โดยกำหนดเงื่อนไขของการฝึกสอนโครงข่ายที่ประสบความสำเร็จ คือ ฝึกสอนจนกระทั่งโครงข่ายมีค่าเฉลี่ยความผิดพลาดกำลังสอง (Mean Square Error : MSE) น้อยกว่าหรือเท่ากับ 0.001 และมีจำนวนรอบการฝึกสอนไม่เกิน 60,000 รอบ (Epoch) และถือว่า การฝึกสอนโครงข่ายที่มีค่าเฉลี่ยความผิดพลาดกำลังสองมากกว่า 0.001 และค่ารอบการฝึกสอนโครงข่ายที่มากกว่า 60,000 รอบ เป็นการฝึกสอนโครงข่ายที่ไม่ประสบความสำเร็จ

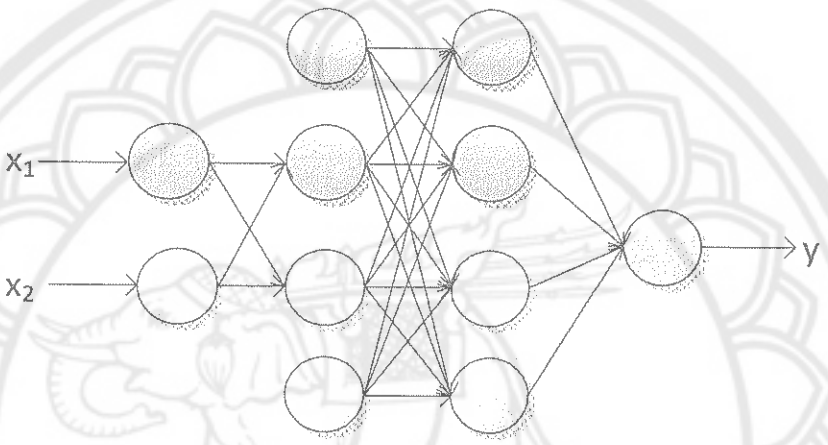
สถาปัตยกรรมโครงข่ายที่ถูกฝึกสอน เรียกย่อๆ ว่า โครงข่าย โดยโครงข่ายที่จะใช้ทดสอบฝึกสอนทั้ง 4 อัลกอริทึมอธิบายได้ดังนี้ โครงข่ายแบบ 3 ชั้น เขียนแทนได้เป็น (a, b, c) และโครงข่ายแบบ 4 ชั้น เขียนแทนได้เป็น (a, b, c, d) เมื่อ a, b, c และ d แทน จำนวนโนดในแต่ละชั้นอธิบายได้ดังภาพ 12 (a) และ 12 (b) แทนโครงข่าย แบบ 3 ชั้น และ 4 ชั้น ตามลำดับ

สำหรับปัญหาและโครงข่ายที่จะทดสอบการฝึกสอนของทั้ง 4 อัลกอริทึม เป็นดังนี้

1. ปัญหา Exclusive-OR โครงข่ายที่ใช้ทดสอบ ได้แก่ (4,1), (4,4,1), (4,4,4,),(4,4,4,1)
2. ปัญหา 3-bit parity โครงข่ายที่ใช้ทดสอบ ได้แก่ (3,1), (5,3,1), (5,3,3,1), (5,3,3,3,1)



(a) (a=2,b=2,c=1)



(b) (a=2,b=4,c=4,d=1)

ภาพ 12 ตัวอย่างของสถาปัตยกรรมของโครงข่าย

หมายเหตุ : (a) โครงข่ายแบบ 3 ชั้น
(b) โครงข่ายแบบ 4 ชั้น

ผลการทดสอบจากการฝึกสอนโครงข่ายของทั้ง 4 อัลกอริทึม แสดงเป็นตาราง แบ่งเป็นคอลัมน์ที่ 1 – คอลัมน์ที่ 7 ซึ่งแต่ละคอลัมน์สามารถอธิบาย ได้ดังนี้ คอลัมน์ 1 แสดงชื่ออัลกอริทึมที่ใช้ฝึกสอนโครงข่าย คอลัมน์ 2 แสดงสถาปัตยกรรมโครงข่ายที่ใช้ฝึกสอนโครงข่าย คอลัมน์ 3 แสดงค่าเวลาเฉลี่ยการฝึกสอนโครงข่ายของแต่ละอัลกอริทึมที่ประสบความสำเร็จ คอลัมน์ 4 แสดงค่ารอบการฝึกสอนโครงข่ายโดยเฉลี่ยของแต่ละอัลกอริทึมที่ประสบความสำเร็จ คอลัมน์ 5 แสดงค่าความสำเร็จการฝึกสอนโครงข่ายของแต่ละอัลกอริทึม คอลัมน์ 6 แสดงจำนวนเวลาที่ใช้ฝึกสอนในแต่ละรอบ (Epoch) และคอลัมน์ที่ 7 แสดงหน่วยความจำที่แต่ละอัลกอริทึมใช้ในการฝึกสอนโครงข่าย

การทดลองที่ 1 เปรียบเทียบประสิทธิภาพการทำงานระหว่างวิธี Least Cross Correlation (LCC) **และ** Levenberg Maquartd (LM)

ผลการทดสอบการฝึกสอนโครงข่ายด้วยอัลกอริทึม LCC, และ LM แสดงดังตาราง 1 และ 2 ตามลำดับ

ตาราง 1 ผลการทำงานของ LCC และ LM ในการแก้ปัญหาเอ็กซ์คูลซีฟออร์ (Exclusive – Or)

1	2	3	4	5	6	7
Algorithm Name	Network #Node	Ave. Time(sec)	Ave. Epoch	Time/Epoch (sec)	Success. (%)	Used Memory. (Byte)
LM	4,1_[5]	0.3	3	0.100	100	41647
LCC	4,1_[5]	0.7	597	0.001	100	4880
LM	4,4,1_[9]	0.4	4	0.100	100	51741
LCC	4,4,1_[9]	1.6	788	0.002	100	7192
LM	4,4,4.4_[13]	0.4	4	0.100	100	61901
LCC	4,4,4.4_[13]	3.6	1309	0.003	100	9496
LM	4,4,4,4,1_[17]	0.4	4	0.100	100	72208
LCC	4,4,4,4,1_[17]	23.6	6814	0.003	100	11800

จากตาราง 1 แสดงให้เห็นว่า การฝึกสอนโครงข่ายของทั้งอัลกอริทึม LCC และ LM ที่ใช้ในการแก้ปัญหาเอ็กซ์คูลซีฟ - ออร์ (Exclusive – Or) นั้น สามารถแก้ปัญหาได้ดี โดยมีค่าความสำเร็จการฝึกสอนโครงข่ายเท่ากับ 100 % โดยไม่ขึ้นกับจำนวนชั้นหรือจำนวนโนดโครงข่าย ส่วนเวลาที่ใช้ในการฝึกสอนโครงข่าย พบว่า การฝึกสอนโครงข่ายด้วยอัลกอริทึม LM ใช้เวลาในการฝึกสอนโครงข่ายน้อยกว่า LCC และยังสอดคล้องกับค่าเฉลี่ยจำนวนรอบที่ใช้ในการฝึกสอนโครงข่าย เนื่องจาก การฝึกสอนโครงข่ายด้วย LM นั้นใช้ค่ารอบเฉลี่ยการฝึกสอนโครงข่ายน้อยกว่าการฝึกสอนโครงข่ายด้วย LCC แต่เมื่อพิจารณาค่าเวลาเฉลี่ยที่ใช้ในฝึกสอนโครงข่ายต่อรอบของการฝึกสอนโครงข่าย พบว่า LCC ใช้เวลาในการประมวลผลต่อรอบน้อยกว่า LM ซึ่งแสดงให้เห็นอย่างชัดเจนว่า ขั้นตอนการประมวลผลของอัลกอริทึม LCC นั้นมีความซับซ้อนน้อยกว่าอัลกอรทึม LM สังเกตได้จากเวลาที่ใช้ในการประมวลผลต่อรอบที่น้อยกว่ามาก เมื่อเปรียบเทียบกับ LM และในเรื่องหน่วยความจำที่ใช้ในการประมวลผลทั้งหมด พบว่า LM นั้น ใช้หน่วยความจำในการประมวลผลน้อยมากเมื่อเปรียบเทียบกับ LM ซึ่งผลการทดลองในตาราง 1

ตาราง 2 ผลการทดสอบการทำงานของอัลกอริทึมในการแก้ปัญหาสามบิตพาริตี้ (3 Bit 3 Parity)

1	2	3	4	5	6	7
Algorithm Name	Network #Node	Ave. Time (sec)	Ave. Epoch	Time/Epoch (sec)	Success. (%)	Used Memory (Byte)
LM	3,1_[4]	2.7	238	0.011	80	41663
LCC	3,1_[4]	18	9195	0.002	100	3592
LM	5,3,1_[9]	2.2	144	0.015	100	51882
LCC	5,3,1_[9]	45.0	12069	0.004	100	5560
LM	5,3,3,1_[12]	1.0	48	0.021	100	61869
LCC	5,3,3,1_[12]	80.0	16377	0.005	100	6856
LM	5,3,3,3,1_[15]	0.4	21	0.033	100	71976
LCC	5,3,3,3,1_[15]	472.0	18892	0.024	60	8152

ในผลการทดลองการแก้ปัญหาสามบิตพาริตี้ (3 Bit Parity) จากตาราง 2 จะเห็นว่าค่าความสำเร็จการฝึกสอนโครงข่ายของ LM นั้นมีค่าเพิ่มขึ้น เมื่อจำนวนชั้นและโนดในโครงข่ายเพิ่มขึ้น สังเกตได้จาก เมื่อโครงข่ายเป็นแบบ 3,1 [4] อัลกอริทึม LM จะมีค่าความสำเร็จในการฝึกสอนโครงข่ายเป็น 80 % แต่เมื่อเพิ่มจำนวนโนดและชั้นในโครงข่ายมากขึ้น ส่งผลให้ค่าความสำเร็จการฝึกสอนโครงข่ายมีค่าเพิ่มขึ้นเป็น 100 % ซึ่งผลการทดลองนี้ ตรงกันข้ามกับ LCC ซึ่งพบว่า LCC นั้นจะทำงานได้ค่าความสำเร็จการฝึกโครงข่ายเป็น 100 % เมื่อจำนวนชั้นในโครงข่ายไม่เกิน 4 ชั้น และจำนวนโนดในโครงข่ายอยู่ระหว่าง 2-12 โนด แต่เมื่อเพิ่มจำนวนโนดในโครงข่ายเป็น 15 โนดและจำนวนชั้นในโครงข่ายเป็น 5 ชั้น พบว่า LCC จะทำงานได้ค่าความสำเร็จการฝึกโครงข่ายลดลงเป็น 60 % ซึ่งแสดงให้เห็นว่า LCC นั้น เหมาะสมสำหรับการฝึกสอนโครงข่ายประสาทเทียมที่มีจำนวนชั้นไม่เกิน 4 ชั้นและจำนวนโนดในโครงข่ายไม่เกิน 15 โนด ส่วนจำนวนรอบเฉลี่ยการฝึกสอนโครงข่ายด้วย LM นั้นมีค่าน้อยกว่าการฝึกสอนโครงข่ายด้วย LCC และยังสอดคล้องกับจำนวนเวลาเฉลี่ยที่ใช้ฝึกสอนโครงข่ายด้วย แต่เมื่อพิจารณาเวลาฝึกสอนโครงข่ายต่อรอบแล้ว พบว่า เวลาเฉลี่ยที่ใช้ฝึกสอนโครงข่ายต่อรอบของ LCC มีค่าน้อยกว่า LM ซึ่งแสดงให้เห็นว่า ถึงแม้ว่า LCC จะใช้จำนวนรอบเฉลี่ยและเวลาเฉลี่ยในการฝึกสอนโครงข่ายมากกว่า LM แต่เมื่อพิจารณาเวลาเฉลี่ยที่ใช้ในในการประมวลต่อรอบของ LCC และ LM แล้ว พบว่า LCC ใช้เวลาเฉลี่ยที่ใช้ในการฝึกสอนโครงข่ายต่อรอบน้อยกว่า LM ซึ่งเป็นผลโดยตรงมาจากขั้นตอนใน

การประมวลผลของ LCC ซึ่งในขั้นตอนการประมวลผลของ LCC นั้นจะไม่มีการคำนวณค่าอนุพันธ์ แต่ใช้หลักการของความสัมพันธ์น้อยสุดเพื่อปรับค่าน้ำหนักโครงข่าย จึงทำให้ขั้นตอนในการฝึกสอนโครงข่ายในแต่ละรอบนั้นใช้เวลาน้อยมาก เมื่อเปรียบเทียบกับอัลกอริทึมอื่นๆ ทั้งยังรวมไปถึงการใช้หน่วยความจำ ซึ่งอัลกอริทึม LCC นั้นใช้หน่วยความจำในการฝึกสอนโครงข่ายน้อยกว่า LM

การทดลองที่ 2 เปรียบเทียบประสิทธิภาพการทำงานระหว่างวิธี Least Cross Cross Correlation (LCC) และ Gradient descent with adaptive learning rule backpropagation (GDA)

ตาราง 3 ผลการทดสอบการทำงานของอัลกอริทึมในการแก้ปัญหาเอ็กซ์คูลซีฟออร์ (Exclusive – Or)

1	2	3	4	5	6	7
Algorithm Name	Network #Node	Ave. Time (sec)	Ave. Epoch	Time/Epoch (sec)	Success. (%)	Used Memory. (Byte)
GDA	4,1_[5]	6.5	895	0.007	100	41465
LCC	4,1_[5]	0.7	597	0.001	100	4880
GDA	4,4,1_[9]	4.7	539	0.009	100	51532
LCC	4,4,1_[9]	1.6	788	0.002	100	7192
GDA	4,4,4.4_[13]	1.3	115	0.011	100	61719
LCC	4,4,4.4_[13]	3.6	1309	0.003	100	9496
GDA	4,4,4,4,1_[17]	1.1	80	0.013	100	72026
LCC	4,4,4,4,1_[17]	23.6	6814	0.003	100	11800

จากผลการทดลองในตาราง 3 การฝึกสอนโครงข่ายโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นลอจิก เอ็กซ์คูลซีฟออร์ (Exclusive – Or) ด้วย GDA และ LCC นั้น พบว่า ค่าความสำเร็จการฝึกสอนโครงข่ายเป็น 100 % ทุกโครงข่าย แสดงว่า ความสำเร็จของการฝึกสอนโครงข่ายนั้นไม่ขึ้นอยู่กับจำนวนโนดหรือชั้นในโครงข่าย อีกทั้งการฝึกสอนโครงข่ายด้วย GDA และ LCC นั้นในโครงข่ายแบบ 4,1 [5] รอบเฉลี่ยการฝึกสอนโครงข่ายด้วย GDA นั้นมีค่าสูงกว่าค่า LCC เล็กน้อย จากนั้น เมื่อจำนวนโนดและชั้นในโครงข่ายเพิ่มขึ้น พบว่า ค่ารอบเฉลี่ยในการฝึกสอนโครงข่ายด้วย

GDA นั้น มีค่าลดน้อยลงจนกระทั่ง ค่ารอบเฉลี่ยการฝึกสอนโครงข่ายมีค่าน้อยกว่าค่ารอบเฉลี่ยการฝึกสอนโครงข่ายด้วย LCC ดังผลการทดลองในตาราง 2 และยังสอดคล้องกับเวลาที่ใช้ในการฝึกสอนโครงข่ายเนื่องจากจำนวนรอบเฉลี่ยมีค่ามาก จะทำให้ค่าเวลาเฉลี่ยการฝึกสอนโครงข่ายมีค่ามากขึ้นด้วย จากผลการทดลองนี้แสดงให้เห็นว่า การฝึกสอนโครงข่ายด้วยข้อมูลฝึกสอน เอ็กซ์คูลซีฟออร์ (Exclusive – Or) โดยอัลกอริทึม GDA นั้นเหมาะสมกับการฝึกสอนโครงข่ายที่มีจำนวนโนดตั้งแต่ 9 โนด และจำนวนชั้นตั้งแต่ 3 ชั้นเป็นต้นไป และเมื่อพิจารณาค่าเวลาเฉลี่ยที่ใช้ฝึกสอนโครงข่ายต่อรอบแล้ว พบว่า LCC ใช้ค่าเวลาเฉลี่ยในการฝึกสอนโครงข่ายต่อรอบน้อยกว่า GDA ซึ่งเป็นผลโดยตรงมาจากขั้นตอนการประมวลผลของ GDA ซึ่งต้องมีการหาค่าอนุพันธ์อันดับหนึ่งเพื่อหาค่าเกรเดียนท์และปรับค่าอัตราการเรียนรู้ (Learning Rate) ในระหว่างการฝึกสอนโครงข่ายซึ่งตรงกันข้ามกับขั้นตอนการประมวลผลของ LCC ซึ่งไม่มีการหาค่าอนุพันธ์ใดๆ แต่อาศัยหลักการความสัมพันธ์น้อยสุดเพื่อปรับค่าน้ำหนักในการฝึกสอนโครงข่ายด้วยหลักการนี้เองส่งผลให้ LCC ใช้หน่วยความจำในการประมวลผลน้อยกว่า GDA อีกเช่นกัน

ตาราง 4 ผลการทดสอบการทำงานของอัลกอริทึมในการแก้ปัญหาสามบิตพาริตี้ (3 Bit Parity)

1	2	3	4	5	6	7
Algorithm Name	Network #Node	Ave. Time(sec)	Ave. Epoch	Time/Epoch (sec)	Success. (%)	Used Memory (Byte)
GDA	3,1_[4]	51.3	3778	0.013	100	41481
LCC	3,1_[4]	18	9195	0.002	100	3592
GDA	5,3,1_[9]	11.2	1434	0.012	100	51700
LCC	5,3,1_[9]	45.0	12069	0.004	100	5560
GDA	5,3,3,1_[12]	9.3	719	0.011	100	61687
LCC	5,3,3,1_[12]	80.0	16377	0.005	100	6856
GDA	5,3,3,3,1_[15]	7.8	696	0.010	100	71749
LCC	5,3,3,3,1_[15]	472.0	18892	0.024	60	8152

จากผลการทดลองในตาราง 4 ในการแก้ปัญหาสามบิตพาริตี้ พบว่า GDA นั้นมีค่าความสำเร็จในการฝึกสอนโครงข่ายเป็น 100 % ทุกโครงข่าย และในส่วนของ LCC นั้นค่าการฝึกสอนโครงข่ายจะมีค่าลดลงเมื่อจำนวนชั้นในโครงข่ายมีค่ามากกว่า 4 ชั้น และจำนวนโนดในโครงข่ายมากกว่า 12 โนด ซึ่งจากผลการทดลองนี้แสดงให้เห็นว่า LCC ไม่เหมาะสมกับ

การฝึกสอนโครงข่ายด้วยชั้นซ่อนมากกว่า 4 ชั้น และจำนวนโนดในโครงข่ายมากกว่า 12 โนด เมื่อใช้ข้อมูลสำหรับการฝึกสอนโครงข่ายเป็นแบบสามบิตพาริตี้ ซึ่งทำให้จำนวนรอบเฉลี่ยและค่าเวลาเฉลี่ยที่ใช้ในการฝึกสอนโครงข่ายของ LCC นั้นมีค่ามากกว่า GDA ส่วนเวลาเฉลี่ยที่ใช้ฝึกสอนโครงข่ายต่อรอบนั้น LCC จะใช้เวลาเฉลี่ยในการฝึกสอนต่อรอบน้อยกว่า GDA จนกระทั่งเมื่อโครงข่ายเป็น 5,3,3,3,1 [15] พบว่า LCC มีค่าความสำเร็จการฝึกสอนโครงข่ายลดลงเป็น 60 % จึงส่งผลให้ LCC นั้น ใช้เวลาเฉลี่ยในการฝึกสอนต่อรอบมากกว่า GDA และ GDA ยังคงมีค่าความสำเร็จในการฝึกสอนโครงข่ายเท่ากับ 100 % ซึ่งจากฝึกสอนโครงข่ายที่มีมาก่อนหน้านั้น LCC จะใช้เวลาเฉลี่ยในการฝึกสอนโครงข่ายต่อรอบน้อยกว่า GDA มาโดยตลอด และยังพบว่า ในการฝึกสอนโครงข่ายโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้นี้ LCC ยังคงใช้หน่วยความจำในการประมวลผลน้อยกว่า GDA เช่นเดียวกับกรณีที่ใช้ข้อมูลฝึกสอนโครงข่ายเป็นเอ็กซ์คูลซีฟออร์ (Exclusive – Or)

การทดลองที่ 3 เปรียบเทียบประสิทธิภาพการทำงานระหว่างวิธี Least Cross Correlation (LCC) และ Gradient descent backpropagation (GD)

ตาราง 5 ผลการทดสอบการทำงานของอัลกอริทึมในการแก้ปัญหาเอ็กซ์คูลซีฟออร์ (Exclusive – Or)

1	2	3	4	5	6	7
Algorithm Name	Network #Node	Ave. Time(sec)	Ave. Epoch	Time/Epoch (sec)	Success. (%)	Used Memory. (Byte)
GD	4,1_[5]	4.1	974	0.004	90	28499
LCC	4,1_[5]	0.7	597	0.001	100	4880
GD	4,4,1_[9]	5.2	1127	0.005	90	35434
LCC	4,4,1_[9]	1.6	788	0.002	100	7192
GD	4,4,4,1._[13]	35.7	6240	0.006	100	42441
LCC	4,4,4.1_[13]	3.6	1309	0.003	100	9496
GD	4,4,4,4,1_[17]	42.9	6836	0.006	100	49441
LCC	4,4,4,4,1_[17]	23.6	6814	0.003	100	11800

จากผลการทดลองในตาราง 5 แสดงผลการฝึกสอนโครงข่ายประสาทเทียมโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นลอจิก เอ็กซ์คูลซีฟออร์ (Exclusive – Or) พบว่าการฝึกสอนโครงข่ายด้วยอัลกอริทึม LCC มีค่าความสำเร็จการฝึกสอนโครงข่ายเป็น 100 % ทุกโครงข่าย แต่การฝึกสอนโครงข่ายด้วย GD เมื่อโครงข่าย เป็น 4,1 [5] และ 4,4,1 [9] จะมีค่าความสำเร็จการฝึกสอนโครงข่ายเท่ากับ 90 % และมีค่าความสำเร็จการฝึกสอนโครงข่ายเพิ่มขึ้นเป็น 100 % เมื่อโครงข่ายที่ถูกฝึกสอนมีจำนวนโนดและจำนวนชั้นในโครงข่ายเพิ่มขึ้น ซึ่งแสดงให้เห็นว่า GD นั้นเหมาะสมกับการฝึกสอนโครงข่ายที่มีจำนวนโนดมากกว่า 9 โนด และชั้นซ่อนมากกว่า 3 ชั้น เป็นต้นไป พิจารณาในคอลัมน์ที่ 3-4 คือ ค่ารอบเฉลี่ยการฝึกสอนโครงข่ายและเวลาเฉลี่ยที่ใช้ฝึกสอนโครงข่าย พบว่า LCC นั้นมีค่าค่ารอบเฉลี่ยการฝึกสอนโครงข่ายและเวลาเฉลี่ยที่ใช้ฝึกสอนโครงข่ายน้อยกว่า GD และเมื่อพิจารณาเวลาเฉลี่ยที่ใช้ในการฝึกสอนโครงข่ายต่อรอบแล้ว พบว่า LCC คงใช้เวลาในการฝึกสอนโครงข่ายต่อรอบน้อยกว่า GD นอกจากนั้น LCC ยังคงใช้หน่วยความจำในการประมวลผลน้อยกว่า GD

ตาราง 6 ผลการทดสอบการทำงานของอัลกอริทึมในการแก้ปัญหาสามบิตพาริตี้ (3 Bit Parity)

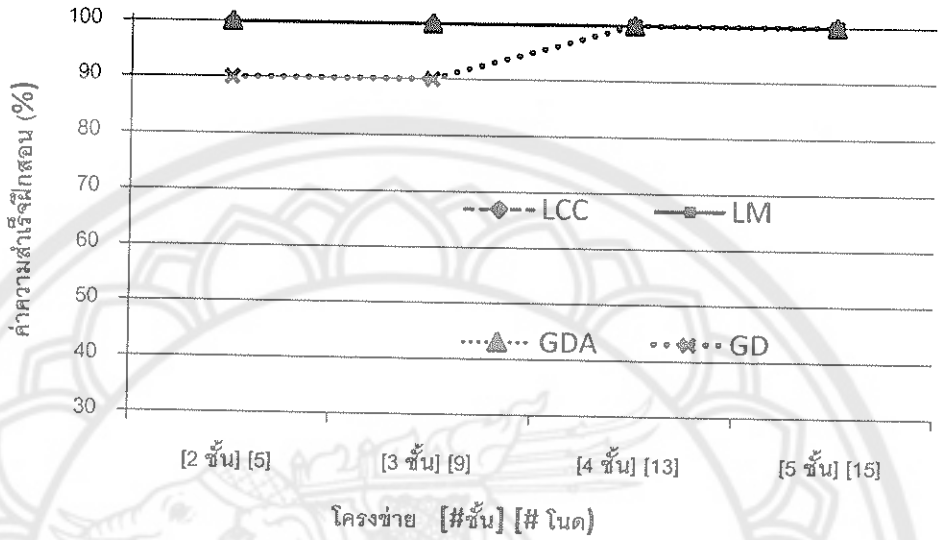
1	2	3	4	5	6	7
Algorithm Name	Network #Node	Ave. Time(sec)	Ave. Epoch	Time/Epoch (sec)	Success. (%)	Used Memory (Byte)
GD	3,1_[4]	100.0	21103	0.005	20	28499
LCC	3,1_[4]	18	9195	0.002	100	3592
GD	5,3,1_[9]	105	18676	0.006	80	35602
LCC	5,3,1_[9]	45.0	12069	0.004	100	5560
GD	5,3,3,1_[12]	94.0	13093	0.007	90	42409
LCC	5,3,3,1_[12]	80.0	16377	0.005	100	6856
GD	5,3,3,3,1_[15]	0.7	9301	0.008	90	49288
LCC	5,3,3,3,1_[15]	472.0	18892	0.024	60	8152

จากผลการทดลองตาราง 6 แสดงการฝึกสอนโครงข่ายประสาทเทียมโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity) พบว่า GD นั้นมีค่าความสำเร็จการฝึกสอนโครงข่ายน้อยมาก มีค่าเพียง 20 % เมื่อโครงข่ายเป็น 3,1 [4] และจะมีค่าความสำเร็จการฝึกสอนโครงข่ายเพิ่มขึ้นเป็น 90-100 % เมื่อจำนวนโนดและจำนวนชั้นในโครงข่ายเพิ่มขึ้น ซึ่งตรงกันข้ามกับ LCC ซึ่งจะมีค่าความสำเร็จการฝึกสอนโครงข่ายลดลงเมื่อจำนวนโนดและจำนวนชั้นในโครงข่ายเพิ่มขึ้นมากๆ ส่วนในคอลัมน์ที่ 3 แสดงค่ารอบเฉลี่ยที่ใช้ในการฝึกสอนโครงข่าย พบว่า LCC จะมีค่ารอบเฉลี่ยการฝึกสอนโครงข่ายที่น้อยเมื่อจำนวนโนดและจำนวนชั้นซ่อนมีค่าน้อย และจะมีค่ารอบเฉลี่ยการฝึกสอนโครงข่ายเพิ่มมากขึ้นเมื่อจำนวนโนดในโครงข่ายเพิ่มขึ้น ซึ่งตรงกันข้ามกับ GD ซึ่งจะมีค่ารอบเฉลี่ยการฝึกสอนโครงข่ายลดน้อยลงเมื่อจำนวนชั้นและจำนวนโนดในโครงข่ายเพิ่มขึ้น ในเรื่องเวลาเฉลี่ยที่ใช้ในการฝึกสอนโครงข่ายต่อรอบนั้น โดยส่วนใหญ่แล้ว LCC จะใช้เวลาในการฝึกสอนโครงข่ายต่อรอบน้อยกว่า GD ยกเว้นในกรณีที่โครงข่ายที่ฝึกสอนนั้นมีจำนวนโนดและชั้นในโครงข่ายมากขึ้น จึงทำให้ LCC ใช้เวลาในการฝึกสอนโครงข่ายต่อรอบมากขึ้น และพบว่า LCC นั้นยังคงใช้เวลาหน่วยความจำในการฝึกสอนโครงข่ายน้อยกว่า GD เหมือนดังหลายๆ กรณีที่ผ่านมา

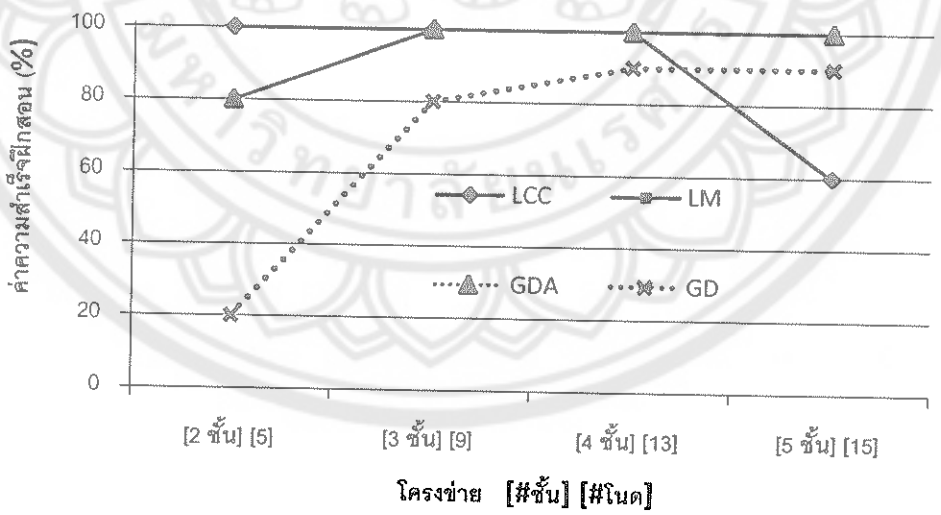
ในส่วนนี้เป็นการเปรียบเทียบผลการทำงานของทั้งสี่อัลกอริทึมซึ่งได้แก่อัลกอริทึม LM, GDA, GD และ LCC ซึ่งแสดงผลการทดสอบไว้ในกราฟเส้นภาพ 13 เป็นการแสดงค่าความสำเร็จในการฝึกสอนโครงข่ายของทั้งสี่อัลกอริทึม จะเห็นว่า ในการฝึกสอนโครงข่ายโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นลอจิก เอ็กซ์คูลซีฟออร์ (Exclusive – Or) ทั้งสี่อัลกอริทึมทำงานได้ดีโดยสังเกตจากค่าความสำเร็จในการฝึกสอนโครงข่ายมีค่าเป็น 100 % ทั้งสี่อัลกอริทึม ถึงแม้ว่า GD นั้นจะมีค่าความสำเร็จในการฝึกสอนโครงข่ายอยู่ที่ 90 % ในช่วงแรก แต่เมื่อเพิ่มจำนวนโนดและชั้นในโครงข่ายขึ้น ก็พบว่า ค่าความสำเร็จการฝึกสอนโครงข่ายมีค่าเป็นเพิ่มเป็น 100 % ซึ่งแสดงให้เห็นว่าการฝึกสอนโครงข่ายโดย GD นั้นเหมาะสมกับการฝึกสอนโครงข่ายที่มีจำนวนชั้นและจำนวนโนดโนดโครงข่ายมากๆ ดังแสดงในภาพ 13

ในภาพ 14 แสดงค่าความสำเร็จการฝึกสอนโครงข่ายโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity) นี้ พบว่า LM มีค่าความสำเร็จในการฝึกสอนโครงข่ายเท่ากับ 100 % ทุกโครงข่าย ส่วนค่าความสำเร็จการฝึกสอนโครงข่ายของ GDA และ GD จะมีค่าน้อยเมื่อจำนวนโนดและชั้นในโครงข่ายมีค่าน้อย และจะมีค่าความสำเร็จในการฝึกโครงข่ายเพิ่มมากขึ้นเมื่อจำนวนโนดและชั้นในโครงข่ายเพิ่มมากขึ้น ซึ่งพบว่า ตรงกันข้ามกับ LCC ซึ่งจากกราฟภาพ 14 แสดงให้เห็นว่า ค่าความสำเร็จในการฝึกสอนโครงข่ายจะมีค่าลดลงเมื่อจำนวนโนดและชั้นใน

โครงข่ายมีค่ามากขึ้น และยังแสดงให้เห็นอีกว่าอัลกอริทึม LCC นั้นไม่เหมาะสมในการประมวลผลสำหรับโครงข่ายที่มีจำนวนโนดและชั้นในโครงข่ายมาก ๆ

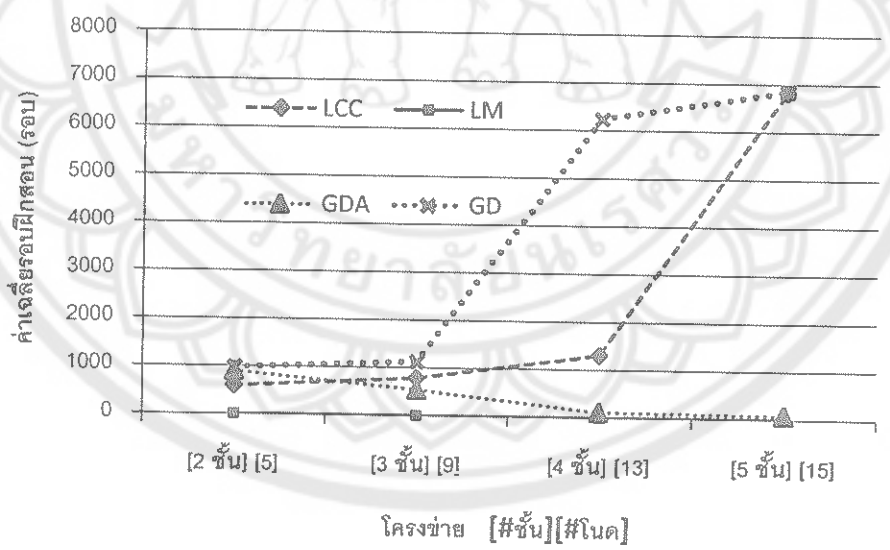


ภาพ 13 ความสัมพันธ์ระหว่างความสำเร็จการฝึกสอน โดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นเอ็กซ์คูลซีฟออร์ (Exclusive – Or)

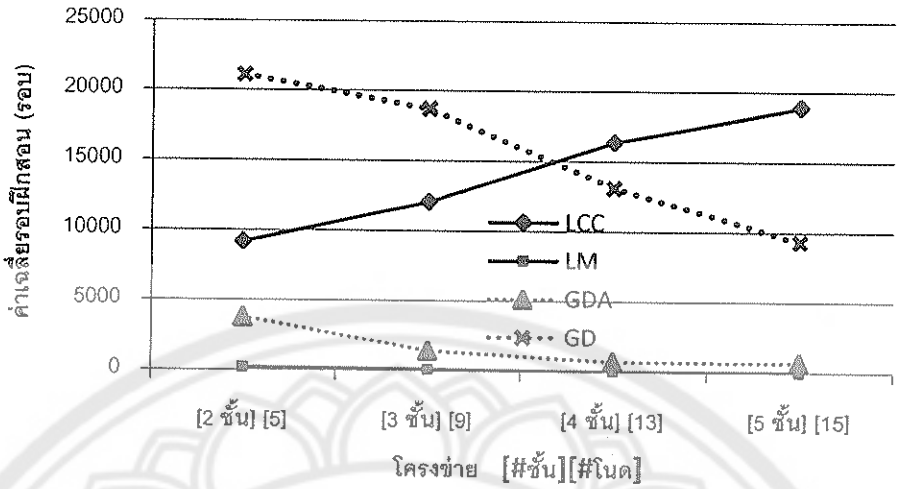


ภาพ 14 ความสัมพันธ์ระหว่างความสำเร็จการฝึกสอนและโครงข่าย โดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity)

ถัดไปในภาพ 15 และภาพ 16 แสดงค่ารอบเฉลี่ยที่ใช้ในการฝึกสอนโครงข่ายที่ใช้ข้อมูลฝึกสอนเป็น เอ็กซ์คูลซีฟออร์ (Exclusive – Or) และสามบิตพาริตี้ (3 Bit Parity) ตามลำดับ ซึ่งทั้งสองกราฟแสดงผลค่อนข้างจะสอดคล้องกัน โดยการฝึกสอนโครงข่ายด้วย LM, GDA นั้นจะมีค่ารอบเฉลี่ยการฝึกสอนโครงข่ายลดลงเมื่อจำนวนโนดและชั้นในโครงข่ายมีค่ามากขึ้น และการฝึกสอนโครงข่ายด้วย LCC นั้น เมื่อจำนวนโนดและชั้นในโครงข่ายมีค่าเพิ่มขึ้น กลับทำให้ค่ารอบเฉลี่ยในการฝึกสอนโครงข่ายมีค่าเพิ่มขึ้น อาจกล่าวได้ว่า ค่ารอบเฉลี่ยของการฝึกสอนโครงข่ายด้วย LCC นั้นมีค่าแปรผันตามจำนวนโนดและจำนวนชั้นในโครงข่าย ยิ่งจำนวนโนดและจำนวนชั้นในโครงข่ายมีค่ามากจะทำให้รอบเฉลี่ยในการฝึกสอนโครงข่ายยิ่งมีค่ามากขึ้นตามไปด้วย ส่วนในกรณีการฝึกสอนโครงข่ายด้วย GD พบว่า ในการฝึกสอนโครงข่ายโดยใช้ข้อมูลฝึกสอนเป็นเอ็กซ์คูลซีฟออร์ (Exclusive – Or) ค่ารอบเฉลี่ยการฝึกสอนโครงข่ายมีค่าเพิ่มขึ้นเมื่อจำนวนโนดและชั้นในโครงข่ายมีค่าเพิ่มขึ้น สอดคล้องกันการทำงานของ LCC และเมื่อเปลี่ยนข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity) ค่ารอบเฉลี่ยการฝึกสอนโครงข่ายกลับมีค่าลดลงเมื่อจำนวนโนดและจำนวนชั้นในโครงข่ายมีค่าเพิ่มขึ้น สอดคล้องกับการฝึกสอนโครงข่ายด้วย LM และ GDA



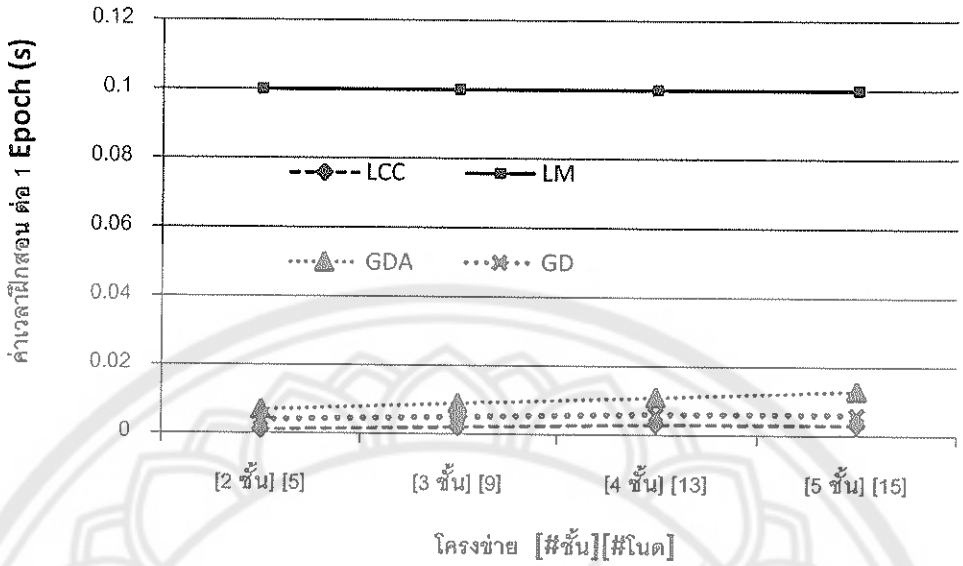
ภาพ 15 ความสัมพันธ์ระหว่างจำนวนรอบเฉลี่ยการฝึกสอนและโครงข่าย โดยใช้ข้อมูลฝึกสอนโครงข่ายเป็น เอ็กซ์คูลซีฟออร์ (Exclusive – Or)



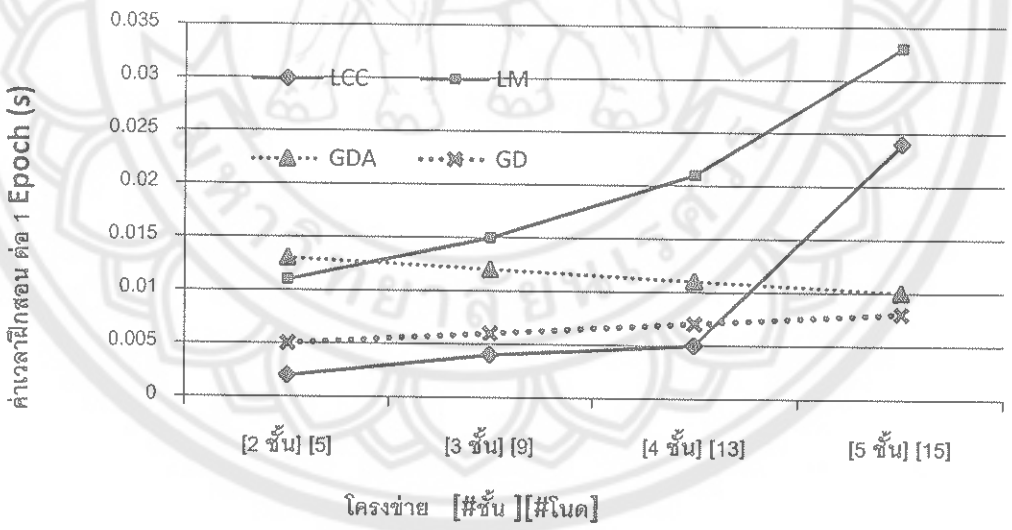
ภาพ 16 ความสัมพันธ์ระหว่างจำนวนรอบเฉลี่ยการฝึกสอนและโครงข่าย โดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity)

ถัดไปในภาพ 17 และภาพ 18 แสดงค่าเวลาเฉลี่ยที่ใช้ในการฝึกสอนโครงข่ายต่อรอบ โดยใช้ข้อมูลฝึกสอนเป็นเอ็กซ์คูลซีฟออร์ (Exclusive – Or) และสามบิตพาริตี้ (3 Bit Parity) ตามลำดับ ผลการทดลองทั้งสองกราฟแสดงสอดคล้องกัน นั่นคือ การฝึกสอนโครงข่ายด้วย LM นั้นใช้เวลาเฉลี่ยในการฝึกสอนโครงข่ายต่อรอบมากที่สุด รองลงมาเป็น GDA, GD และ LCC ตามลำดับ และทั้งสี่อัลกอริทึมจะใช้เวลาการฝึกสอนโครงข่ายต่อรอบเพิ่มขึ้นเมื่อจำนวนโนดและชั้นในโครงข่ายเพิ่มมากขึ้น เห็นได้อย่างชัดเจนดังแสดงในภาพ 18 โดยเฉพาะการฝึกสอนโครงข่ายด้วย LM และ LCC

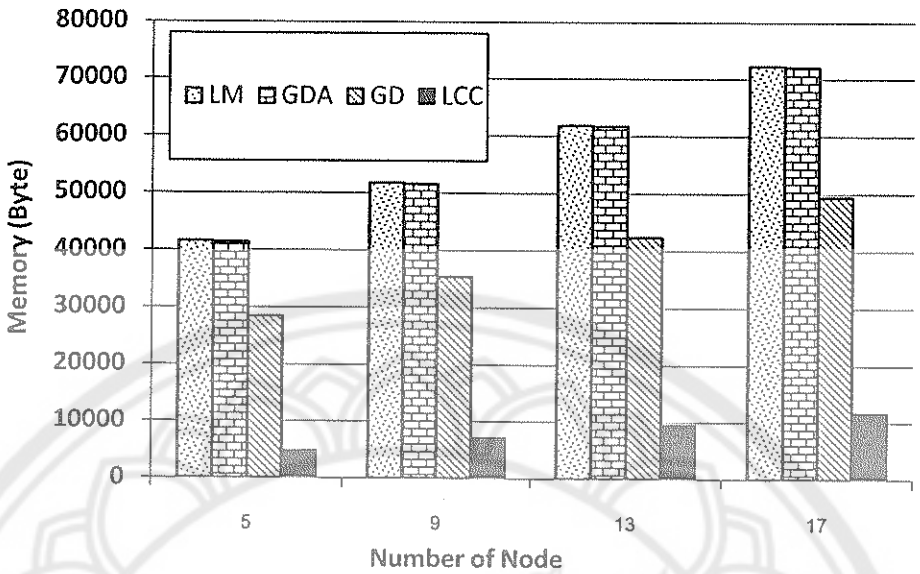
จากกราฟภาพ 19 ถึง 20 แสดงหน่วยความจำที่ใช้ในการฝึกสอนโครงข่ายของแต่ละอัลกอริทึม พบว่า การฝึกสอนโครงข่ายด้วยอัลกอริทึม LCC นั้น ใช้หน่วยความจำน้อยที่สุด และการฝึกสอนโครงข่ายด้วย LM ใช้หน่วยความจำมากสุด รองลงมา คือ GDA และ GD ตามลำดับ ทั้งการฝึกสอนโครงข่ายด้วยปัญหา Exclusive-Or และ 3 Bit Parity และยังพบว่าหน่วยความจำที่ใช้ในการฝึกสอนโครงข่ายจะมีค่าแปรผันตามจำนวนชั้นและจำนวนโนดในโครงข่าย ยิ่งมีจำนวนชั้นและจำนวนโนดในโครงข่ายเพิ่มขึ้น ย่อมทำให้การฝึกสอนโครงข่ายใช้หน่วยความจำในการฝึกสอนเพิ่มขึ้น



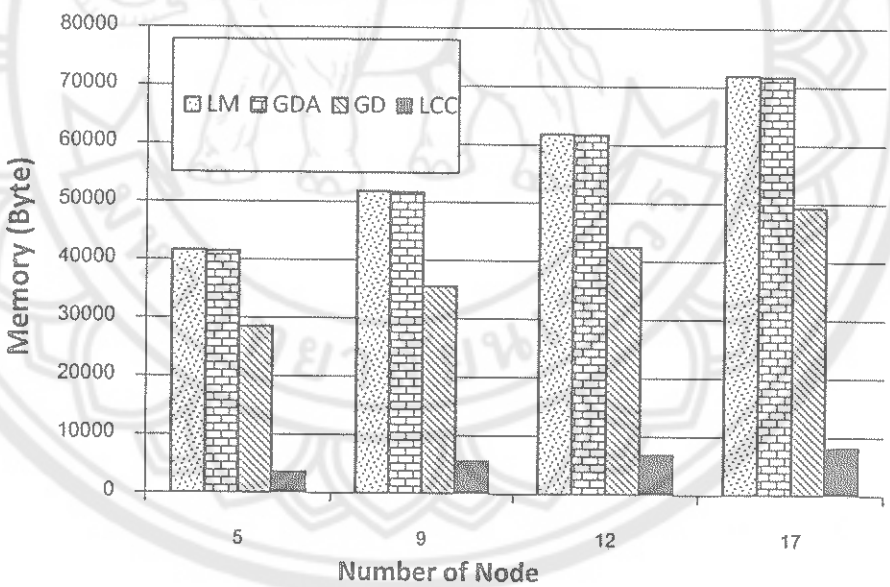
ภาพ 17 ความสัมพันธ์ระหว่างเวลาในการฝึกสอนโครงข่ายแต่ละรอบและโครงข่าย โดยใช้ข้อมูลฝึกสอนโครงข่ายเป็น เอ็กซ์คูลซีฟออร์ (Exclusive – Or)



ภาพ 18 ความสัมพันธ์ระหว่างเวลาในการฝึกสอนโครงข่ายต่อรอบและโครงข่าย โดยข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity)



ภาพ 19 ความสัมพันธ์ระหว่าง หน่วยความจำที่ใช้ในการประมวลผลของแต่ละอัลกอริทึม และจำนวนโนดโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นเอ็กซ์คูลซีฟออร์ (Exclusive – Or)



ภาพ 20 ความสัมพันธ์ระหว่าง หน่วยความจำที่ใช้ในการประมวลผลของแต่ละอัลกอริทึม และจำนวนโนดโดยใช้ข้อมูลฝึกสอนโครงข่ายเป็นสามบิตพาริตี้ (3 Bit Parity)

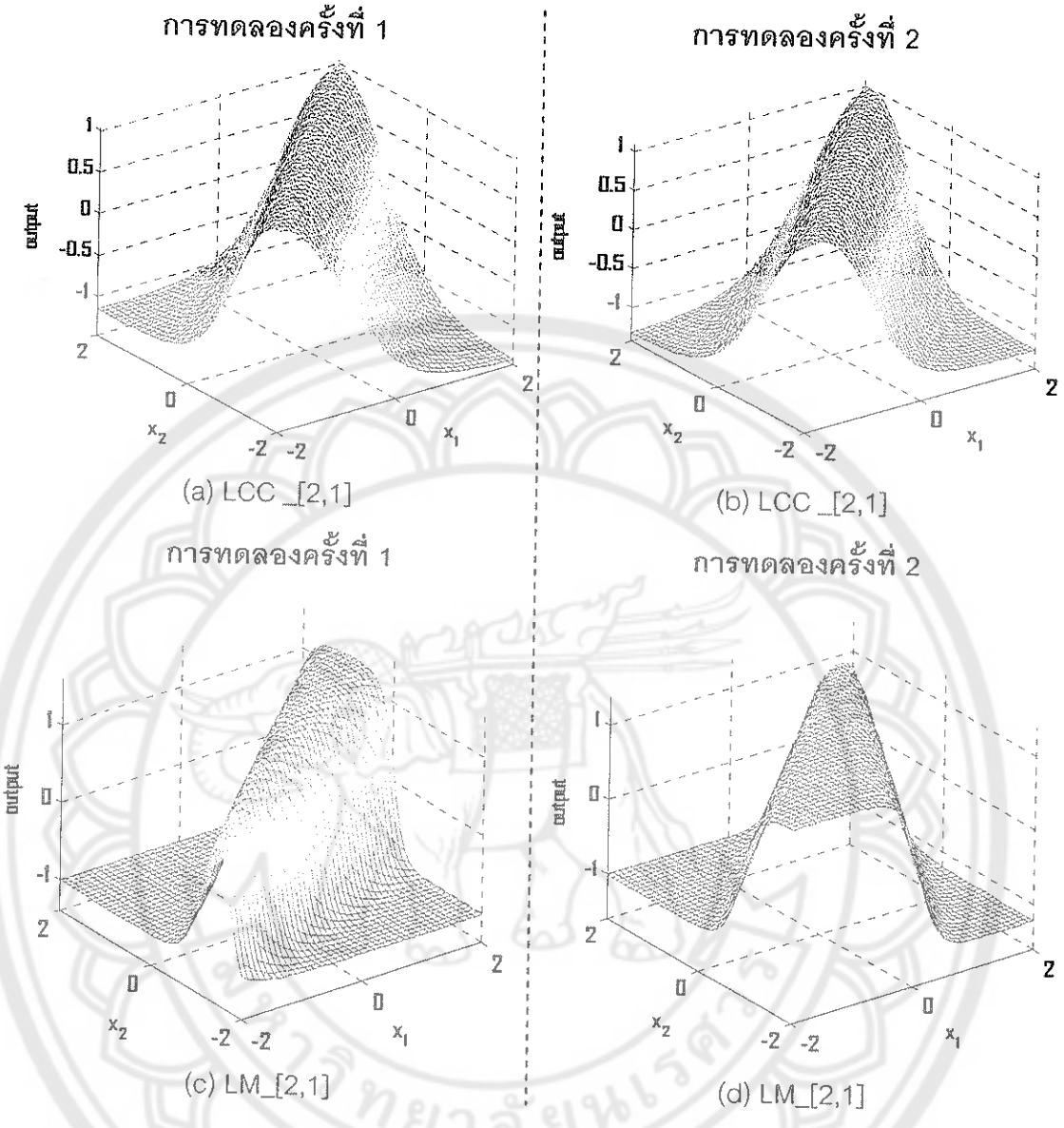
จากการวิเคราะห์ผลการทดสอบการฝึกสอนโครงข่ายของทั้ง 4 อัลกอริทึมที่ผ่านมาโดยทั่วไปแล้ว พบว่า อัลกอริทึม LCC นั้น มีค่าความสำเร็จการฝึกสอนโครงข่ายต่ำกว่าอัลกอริทึม LM และค่าความสำเร็จการฝึกสอนโครงข่ายมีค่าลดลง เมื่อจำนวนชั้นและโนดในโครงข่ายเพิ่มขึ้น เพื่อพิจารณาปัญหาดังกล่าว ได้ทดลองพิจารณาเอาท์พุทที่ได้จากการฝึกสอนโครงข่ายด้วยอัลกอริทึม LM, GDA, GD และ LCC ในการแก้ปัญหา Exclusive–Or ที่มีค่าอินพุทฝึกสอนและเอาท์พุทฝึกสอน ดังตาราง 7

ตาราง 7 ค่าอินพุทและเอาท์พุทของปัญหา Exclusive-Or

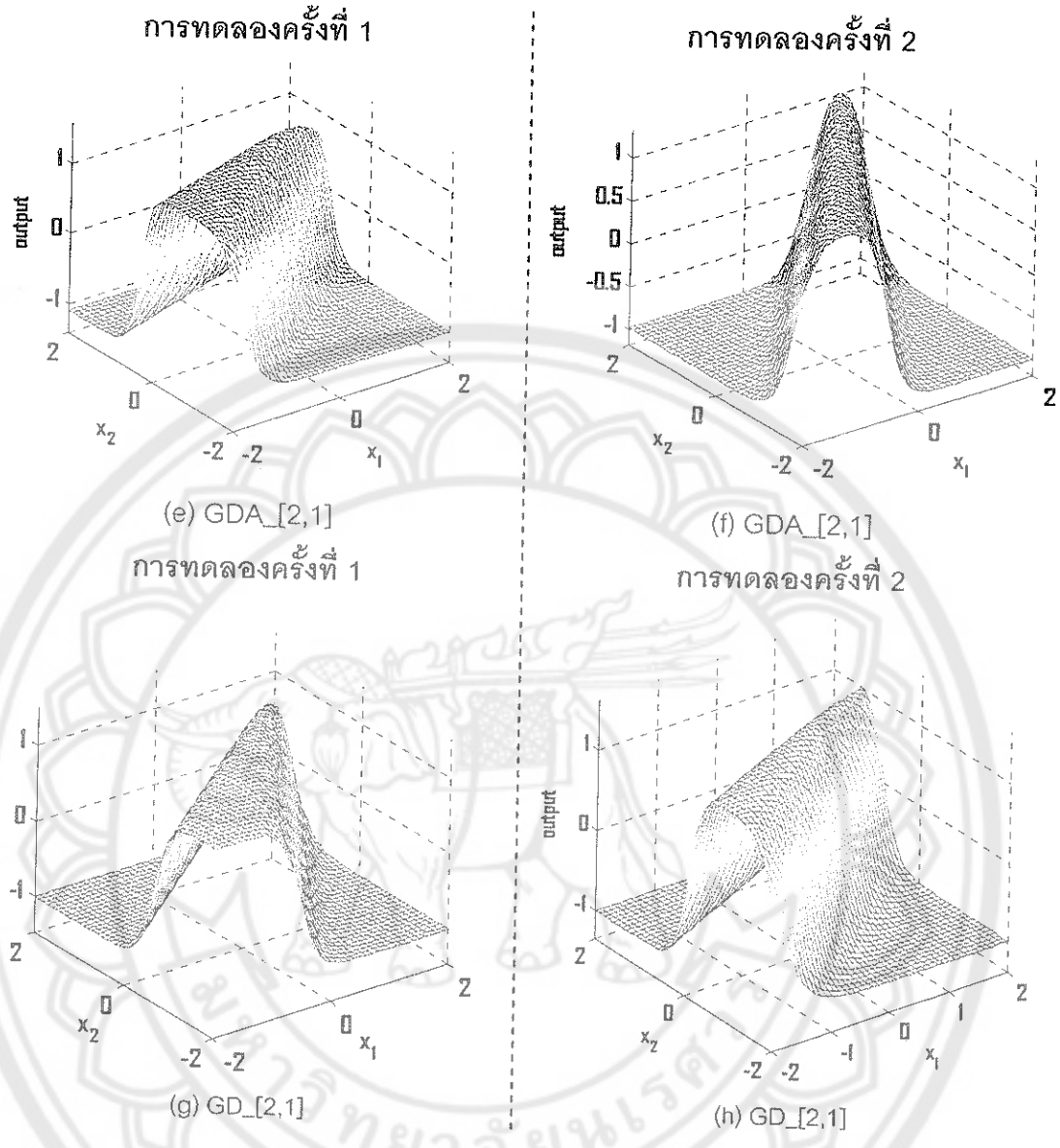
Input X_1	Input X_2	Output y
-1	-1	1
-1	1	-1
1	-1	-1
1	1	1

ตาราง 7 แสดงความสัมพันธ์ระหว่างกันของค่าเอาท์พุทที่ต้องการ และอินพุทฝึกสอนโครงข่าย ซึ่งมีค่าเท่ากับ -1 และ 1 เท่านั้น ตามที่แสดงในตาราง 7 เพื่อทดสอบความถูกต้องและพิจารณาพฤติกรรม หลังจากโครงข่ายถูกฝึกสอนจากอัลกอริทึม LM, GDA, GD และ LCC แล้ว จะกำหนดให้อินพุททดสอบ คือ x_1 และ x_2 มีค่าเท่ากับ [-2,2] เข้าสู่โครงข่ายที่ถูกฝึกสอนด้วยอัลกอริทึม LM, GDA, GD และ LCC มาก่อนหน้านั้นแล้ว โดยใช้แก้ปัญหา Exclusive-Or ผลการทดสอบได้แสดงในภาพ 21 – 23 มีลักษณะเป็น 3 มิติ โดยแกนแนวระนาบ คือค่าอินพุททดสอบ x_1 และ x_2 แกนแนวตั้งเป็นค่าเอาท์พุท y ซึ่งได้จากค่าอินพุททดสอบ ดังแสดงภาพ 21 – 23 (a), (b),(c), (d),(e),(f),(g) และ (h) ตามลำดับ ในแต่ละภาพ จะแสดงผลการฝึกสอนโครงข่าย จำนวน 2 ครั้ง โดยที่โครงข่ายแต่ละโครงข่ายนั้นมีค่าน้ำหนักเริ่มต้นในโครงข่ายต่างกัน

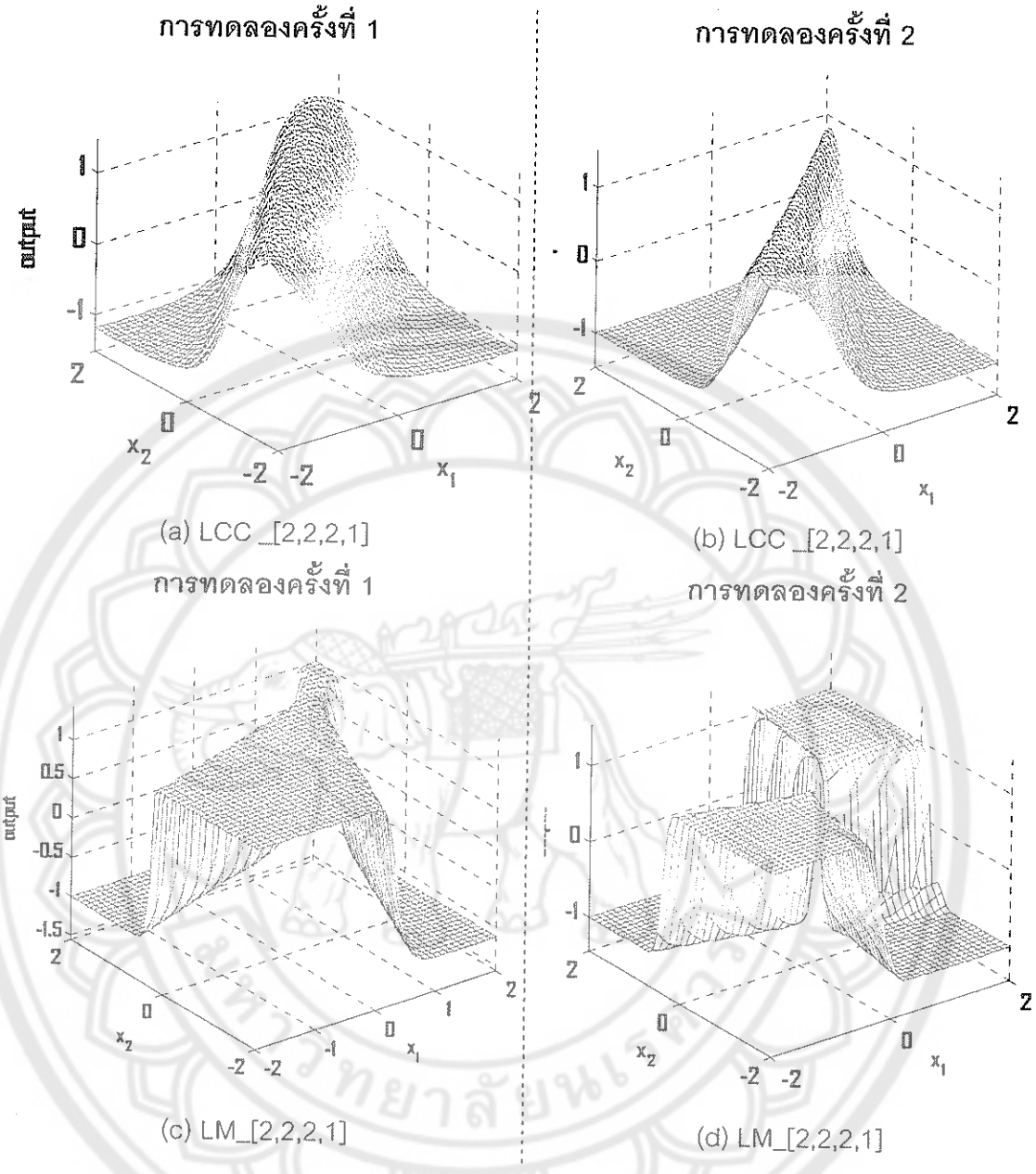
จากภาพ 21 (a), 21(b), 22(a), 22(b), 23(a), และ 23(b) รูปที่เกิดจากการฝึกสอนโครงข่ายด้วยอัลกอริทึม LCC แสดงถึงความสมมาตรที่มั่นคง (Consitency) มีลักษณะการให้คำตอบที่เป็นไปในทิศทางเดียวกันในจุดที่ไม่มีข้อมูลฝึกสอนและแนวโน้มผลลัพธ์มีที่ฝึกสอนนั้นออกมาในลักษณะเดียวกันแม้ว่าค่าน้ำหนักเริ่มต้นต่างกันและโครงข่ายที่ต่างกัน



ภาพ 21 พื้นผิวเอาท์พุทโดยใช้ของข้อมูลทดสอบ x_1 และ x_2 มีค่าเท่ากับ [-2,2] โดยมีโครงข่าย [2,1] ของอัลกอริทึม LCC และ LM

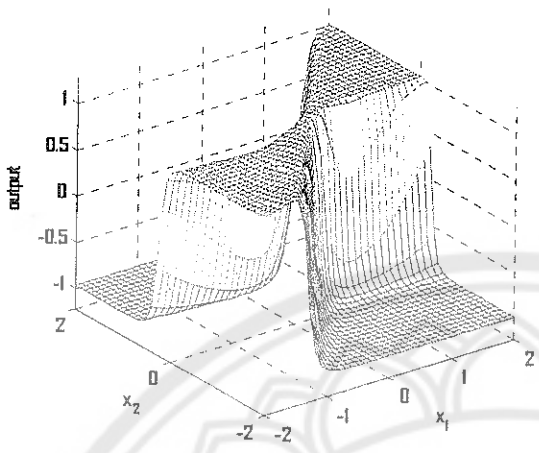


ภาพ 22 พื้นผิวเอาท์พุทโดยใช้ของข้อมูลทดสอบ x_1 และ x_2 มีค่าเท่ากับ [-2,2] โดยมีโครงข่าย [2,1] ของอัลกอริทึม GDA และ GD



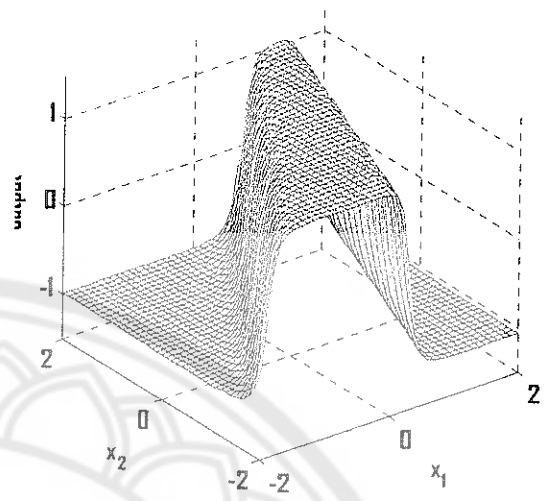
ภาพ 23 พื้นผิวเอาท์พุทโดยใช้ของข้อมูลทดสอบ x1 และ x2 มีค่าเท่ากับ [-2,2] โดยมีโครงข่าย [2,2,2,1] ของอัลกอริทึม LCC และLM

การทดลองครั้งที่ 1



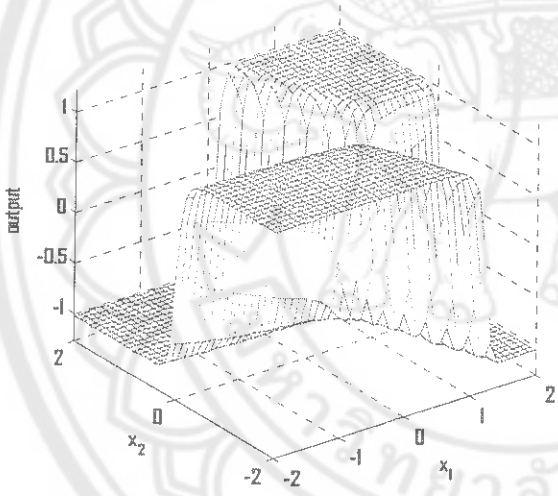
(e) GDA_[2,2,2,1]

การทดลองครั้งที่ 2



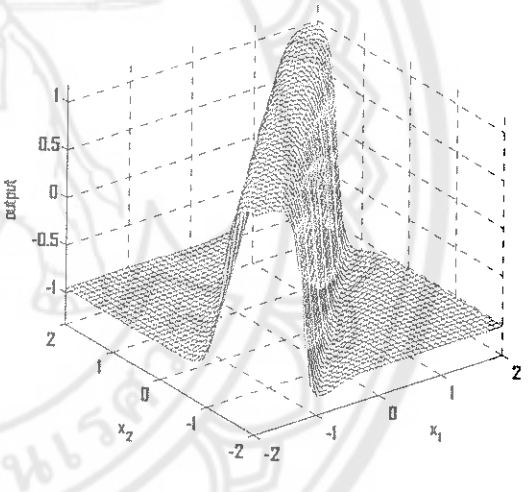
(f) GDA_[2,2,2,1]

การทดลองครั้งที่ 1



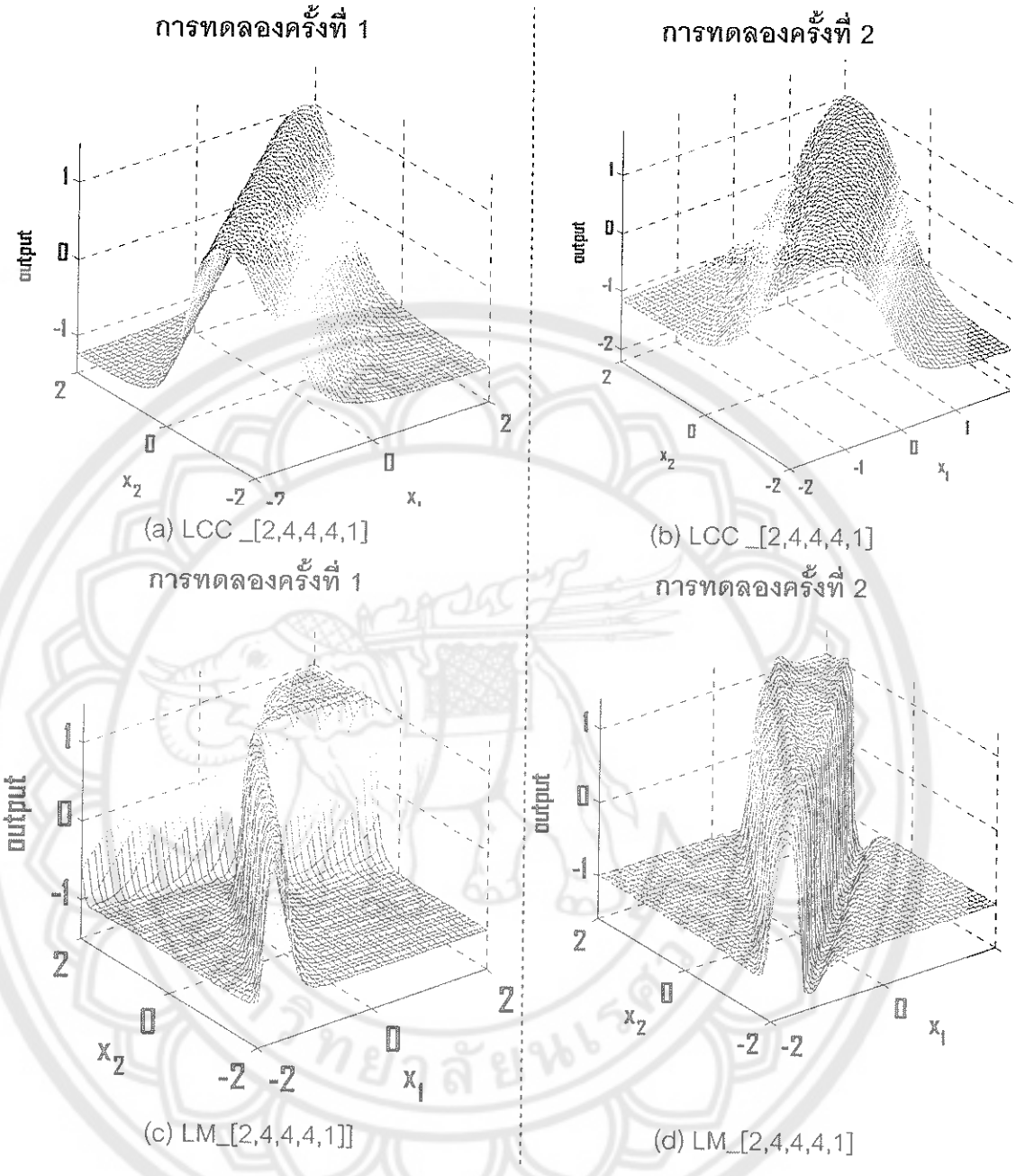
(g) GD_[2,2,2,1]

การทดลองครั้งที่ 2



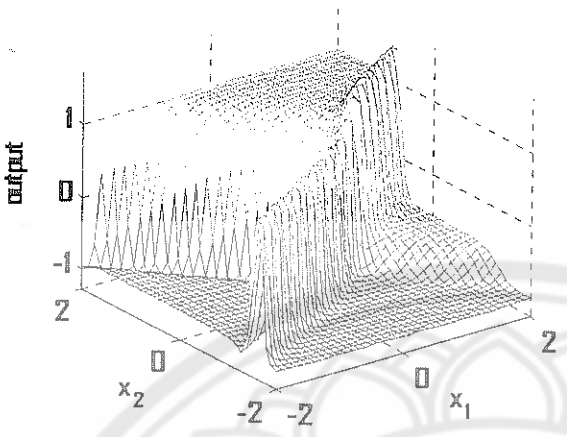
(h) GD_[2,2,2,1]

ภาพ 24 พื้นผิวเอาท์พุทโดยใช้ของข้อมูลทดสอบ x_1 และ x_2 มีค่าเท่ากับ [-2,2] โดยมีโครงข่าย [2,2,2,1] ของอัลกอริทึม GDA และ GD



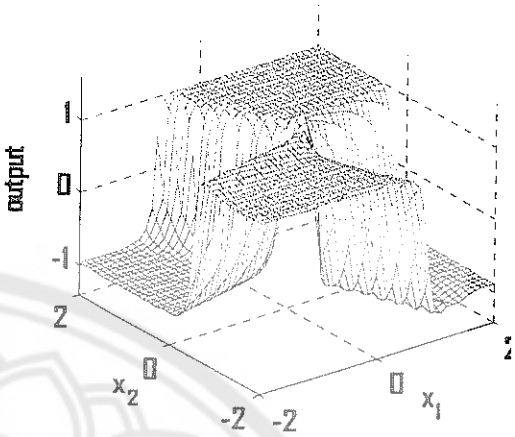
ภาพ 25 พื้นผิวเอาท์พุทโดยใช้ของข้อมูลทดสอบ x1 และ x2 มีค่าเท่ากับ [-2,2] โดยมีโครงข่าย [2,4,4,4,1] ของอัลกอริทึม LC และ LM

การทดลองครั้งที่ 1



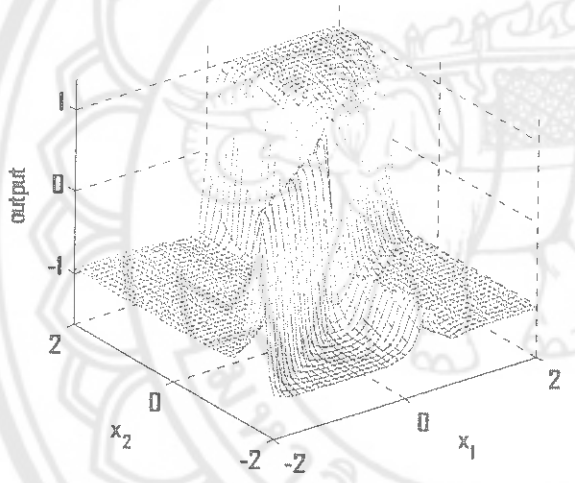
(e) GDA_[2,4,4,4,1]

การทดลองครั้งที่ 2



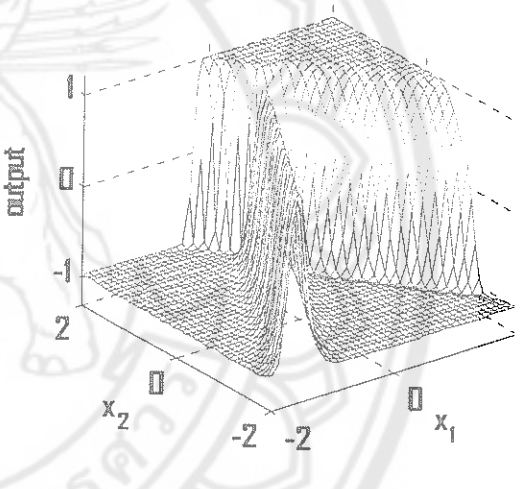
(f) GDA_[2,4,4,4,1]

การทดลองครั้งที่ 1



(g) GD_[2,4,4,4,1]

การทดลองครั้งที่ 2



(h) GD_[2,4,4,4,1]

ภาพ 26 พื้นผิวเอาท์พุทโดยใช้ของข้อมูลทดสอบ x_1 และ x_2 มีค่าเท่ากับ [-2,2] โดยมีโครงข่าย [2,4,4,4,1] ของอัลกอริทึม GDA และ GD