



ใบรับรองวิทยานิพนธ์
บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมเครื่องกล)

ปริญญา

วิศวกรรมเครื่องกล

วิศวกรรมเครื่องกล

สาขา

ภาควิชา

เรื่อง การควบคุมแรงบิดของเครื่องขันสกรูอัตโนมัติโดยเทคนิคโครงข่ายประสาทเทียม

Torque Control of Auto-Screwdriver Machine by Neural Network

นามผู้วิจัย นางสาวกฤติยา พาอิม

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(อาจารย์ชัยยากร จันทร์สุวรรณ, Ph.D.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(อาจารย์คุณยุต เอี่ยมสอาด, Ph.D.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(อาจารย์ชนะ รักศิริ, D.Eng.)

หัวหน้าภาควิชา

(รองศาสตราจารย์ชวลิต กิตติชัยการ, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

(รองศาสตราจารย์กัญจนา วีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ เดือน พ.ศ.

วิทยานิพนธ์

เรื่อง

การควบคุมแรงบิดของเครื่องขันสกรูอัตโนมัติโดยเทคนิคโครงข่ายประสาทเทียม

Torque Control of Auto-Screwdriver Machine by Neural Network

โดย

นางสาวกฤติยา พาอิม

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมเครื่องกล)

พ.ศ. 2555

กฤติยา พานิธิ 2555: การควบคุมแรงบิดของเครื่องขันสกรูอัตโนมัติโดยเทคนิคโครงข่ายประสาทเทียม ปรินญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมเครื่องกล) สาขาวิศวกรรมเครื่องกล ภาควิชาวิศวกรรมเครื่องกล อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก: อาจารย์ชัยยากร จันทร์สุวรรณ, Ph.D. 130 หน้า

งานวิจัยนี้ได้นำเสนอการประยุกต์ใช้หลักการและทฤษฎีของโครงข่ายประสาทเทียมในการควบคุมการขันสกรู เพื่อหาแบบควบคุมให้กับเครื่องขันสกรูอัตโนมัติ โดยประกอบด้วยโครงข่ายประสาทเทียม 2 ส่วน คือ ส่วนของการหาเอกลักษณ์ และส่วนตัวควบคุม ซึ่งใช้หลักการของโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ และมีการเรียนรู้แบบชี้แนะเพื่อทำการปรับเปลี่ยนค่าน้ำหนักและค่าไบแอสของโครงข่ายเพื่อให้ผลลัพธ์ของแบบจำลองใกล้เคียงกับค่าความจริงมากที่สุด โดยทำการแบ่งการเรียนรู้ของโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ ออกเป็น 20 รูปแบบ และการแบ่งการเรียนรู้ของโครงข่ายประสาทที่ใช้เป็นตัวควบคุม ออกเป็น 14 รูปแบบ แล้วทดลองหาจำนวนโหนดในชั้นซ่อนในรูปแบบที่แตกต่างกับเพื่อออกแบบโครงข่ายประสาทเทียมที่ทำให้ค่าความคลาดเคลื่อนต่ำสุด รูปแบบฟังก์ชันการถ่ายโอนของทั้งสองโครงข่ายประสาทในชั้นซ่อนและฟังก์ชันการถ่ายโอนในชั้นเอาต์พุตเป็นลอจิก-ซิกมอยด์และลิเนียร์ตามลำดับ โดยค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ส่วนของการหาเอกลักษณ์ คือ 0.3963 นิวตันเมตร และค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ส่วนเป็นตัวควบคุม คือ 0.0183 นิวตันเมตร โดยการทดสอบการควบคุมระบบซึ่งนำโครงข่ายประสาทเทียมทั้งสองไปใช้นั้นมีค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตเท่ากับ -0.009 นิวตันเมตร ซึ่งอยู่ในขอบเขตที่กำหนด

ลายมือชื่อนิติ

ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

Krittiya Pa-im 2012: Torque Control of Auto-Screwdriver Machine by Neural Network.
Mater of Engineering (Mechanical Engineering), Major Field: Mechanical Engineering,
Department of Mechanical Engineering. Thesis Advisor: Mr. Chaiyakron Chansuwan,
Ph.D. 130 pages.

This research presents an application of the principles and the theory of neural networks in torque control of the screw for the Auto-Screwdriver Machine. The neural network consists of two parts: Neural Network Identification and Neural Network Control. The concept of artificial neural network model for the reverse and learning to adjust the weights and the bias of network for the output of the model close to the target. The learning of neural network identification is 20 nodes and the learning of neural networks control are 14 nodes for the minimize mean square error. The transfer function of the neural network hidden layer and output layer transfer function is log-sigmoid and linear, respectively. The mean square error of the learning process of identification is 0.3963 Nm and the mean error square of the learning process of the controller is 0.0183 Nm. The testing control system, which the neural network both using the error value between the reference signal with the output equal to -0.009 Nm, which is in the control limit.

Student's signature

Thesis Advisor's signature

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีจากความช่วยเหลืออย่างดียิ่งจากหลายๆฝ่าย ขอขอบพระคุณ ดร.ชัยากร จันท์สุวรรณ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก ดร.คุณยุต เอี่ยมสอาด และ ดร.ชนะ รัชศิริ อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ผู้ให้คำแนะนำ ข้อคิดเห็นต่างๆ รวมถึงแนวทางในการแก้ไขปัญหาในการทำวิจัยมาโดยตลอด รวมทั้งคณะกรรมการในการสอบปากเปล่า ขั้นสุดท้ายที่เสียสละเวลาในการคุมสอบและให้คำแนะนำเพื่อให้วิทยานิพนธ์เล่มนี้เสร็จสมบูรณ์

ขอขอบคุณเจ้าหน้าที่ พี่ๆวิศวกร ฝ่าย Tooling ฝ่าย Software ที่กรุณาให้ความร่วมมือเป็นอย่างดี ในการสนับสนุนข้อมูลในงานวิจัยและอำนวยความสะดวก เอื้อเฟื้อสถานที่ในการทำวิจัยครั้งนี้ รวมถึงการให้คำปรึกษาและคำแนะนำต่างๆเกี่ยวกับกระบวนการขึ้นสกรู ขอขอบคุณภาควิชา วิศวกรรมเครื่องกล เจ้าหน้าที่ทุกคนที่อำนวยความสะดวกในการทำงานวิจัยครั้งนี้ ขอขอบคุณพี่ๆ เพื่อนๆ นิสิตทุกคนที่ศึกษาและทำวิจัยด้วยกันมาตลอดมาให้ความช่วยเหลือเสนอแนะข้อคิด และกำลังใจมาตลอด

ท้ายนี้ผู้วิจัยขอกราบขอบพระคุณบิดา มารดา ญาติพี่น้อง ที่สนับสนุนการทำงานวิจัยและคอยมอบกำลังใจด้วยดีเสมอมา ขอขอบคุณทุกท่านที่ไม่ได้กล่าวถึง ณ ที่นี้ ที่ให้ความช่วยเหลือและกำลังใจจนสำเร็จการศึกษา

กฤติยา พาวิ้ม
มีนาคม 2555

สารบัญ

หน้า

สารบัญ	(1)
สารบัญตาราง	(2)
สารบัญภาพ	(3)
คำอธิบายสัญลักษณ์และคำย่อ	(7)
คำนำ	1
วัตถุประสงค์	3
การตรวจเอกสาร	4
อุปกรณ์และวิธีการ	46
อุปกรณ์	46
วิธีการ	46
ผลและวิจารณ์	64
สรุปและข้อเสนอแนะ	90
สรุป	90
ข้อเสนอแนะ	91
เอกสารและสิ่งอ้างอิง	92
ภาคผนวก	95
ภาคผนวก ก โปรแกรมเมทแลป (Matlab) ที่ใช้ในการออกแบบระบบควบคุมการ	
ชันสกรูอัตโนมัติโดยโครงข่ายประสาทเทียม	96
ภาคผนวก ข กราฟแสดงการเรียนรู้ของโครงสร้างของโครงข่ายประสาทเทียมใน	
การหาเอกลักษณ์	111
ภาคผนวก ค กราฟแสดงการเรียนรู้ของโครงสร้างของโครงข่ายประสาทเทียมใน	
การควบคุม	122
ประวัติการศึกษา และการทำงาน	130

สารบัญตาราง

ตารางที่	หน้า
1 แสดงผลการฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์	52
2 แสดงลักษณะ โครงสร้างของโครงข่ายประสาทเทียมในการหาเอกลักษณ์	53
3 แสดงผลการฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์	56
4 แสดงลักษณะ โครงสร้างของโครงข่ายประสาทเทียมในการควบคุม	57
5 คำนำน้หนักของชั้นรับข้อมูลไปยังชั้นซ่อนของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสม	66
6 น้ำหนักของชั้นซ่อนไปยังชั้นเอาต์พุตของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสม	72
7 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสม	78
8 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมที่เหมาะสม	83
9 คำนำน้หนักของชั้นรับข้อมูลไปยังชั้นซ่อนของโครงข่ายประสาทเทียมในการหาระบบควบคุมที่เหมาะสม	85
10 น้ำหนักของชั้นซ่อนไปยังชั้นเอาต์พุตของโครงข่ายประสาทเทียมในการหาระบบควบคุมที่เหมาะสม	87
11 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมในการหาระบบควบคุมที่เหมาะสม	88
12 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมที่เหมาะสม	89

สารบัญภาพ

ภาพที่	หน้า
1 ส่วนประกอบหลัก ภายในฮาร์ดดิสก์ไครฟ์	5
2 แผ่นแพลตเตอร์ หรือแผ่นมีเดียที่ใช้บันทึกข้อมูล	6
3 สปินเคิลมอเตอร์	7
4 แสดงลักษณะหัวอ่านเขียนข้อมูล	8
5 แสดงส่วนประกอบต่างๆ ของแอกชูเอเตอร์ (Actuator)	9
6 แสดงแผงวงจรของฮาร์ดดิสก์	10
7 OUTPUT และ INPUT ของระบบควบคุม	11
8 ส่วนที่เป็นตัวควบคุมและส่วนประกอบต่างๆ	13
9 สัญญาณเข้าและออกของ Plant	13
10 Plant ของระบบควบคุมที่มีการรบกวนจากภายนอก	14
11 ระบบควบคุมแบบวงจรถูกเปิด	15
12 ระบบควบคุมแบบวงจรถูกปิด	16
13 นิเวศน์ที่มีอินพุตเพียงอินพุตเดียว โดยไม่นำค่าไปแอสมาทิด	18
14 นิเวศน์ที่มีหลายอินพุต	20
15 นิเวศน์ที่มีอินพุต R อินพุตแบบย่อ	20
16 ลักษณะการเรียนรู้แบบมีการสอน (Supervised Learning)	22
17 แสดงการเรียนรู้แบบไม่มีการสอน (Unsupervised Learning)	22
18 แสดงรูปแบบ Back- propagation neural network	23
19 แสดงลำดับการประมวลผลของโครงข่าย	30
20 ตัวควบคุมโครงข่ายประสาทโดยตรง	34
21 System Identification Model	35
22 แผนแสดงการควบคุมแบบปรับตัวเองโดยใช้โครงข่ายประสาทเทียม	36
23 แสดงการประยุกต์โครงข่ายประสาทในการหาเอกลักษณ์	37
24 การฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุม	39
25 แผนภาพของระบบควบคุมโดยใช้โครงข่ายประสาทเทียม	42
26 ระบบควบคุมกระบวนการขั้นสูงปัจจุบัน	47

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
27 ระบบควบคุมที่ใช้ในงานวิจัยนี้	48
28 แสดงขั้นตอนการใช้งานโครงข่ายประสาทเทียมในโปรแกรมเมทแลป	49
29 ชุดข้อมูลอินพุตที่ใช้ฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์	51
30 ครรชนิสมรรถนะขณะเรียนรู้ของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่ จำนวนโหนด เท่ากับ 150	54
31 ชุดข้อมูลสำหรับใช้ฝึกโครงข่ายประสาทเทียมในการควบคุม	55
32 ครรชนิสมรรถนะขณะเรียนรู้ของโครงข่ายประสาทเทียมในการควบคุมที่จำนวน โหนดเท่ากับ 30	57
33 แสดงผลตอบสนองของระบบควบคุมที่ปรับตัวอัตโนมัติโดยใช้โครงข่ายประสาทเทียม	58
34 แสดงค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ออกจากระบบ	59
35 แสดงสัญญาณควบคุมจากโครงข่ายประสาทเทียม	59
36 แสดงค่าความผิดพลาดระหว่างโครงข่ายประสาทเทียมเอกลักษณ์และพลานท์	60
37 แสดงผลตอบสนองของระบบควบคุมที่ปรับตัวอัตโนมัติโดยใช้โครงข่ายประสาทเทียม	61
38 แสดงค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ออกจากระบบ	61
39 แสดงสัญญาณควบคุมจากโครงข่ายประสาทเทียม	62
40 แผนภูมิแสดงขั้นตอนการดำเนินงานวิจัย	63
41 ค่าครรชนิสมรรถนะของผลการเรียนรู้รูปของโครงข่ายประสาทเทียมในการหาเอกลักษณ์	65
42 ผลการทดสอบโครงข่ายประสาทเทียมในการหาเอกลักษณ์	66
43 ค่าครรชนิสมรรถนะของผลการเรียนรู้รูปของโครงข่ายประสาทเทียมในการหา ระบบควบคุม	84
44 ผลการทดสอบโครงข่ายประสาทเทียมในการควบคุม	85

สารบัญภาพ (ต่อ)

[illegible]

สารบัญภาพ (ต่อ)

ภาพผนวกที่	หน้า
ค7 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 7	126
ค8 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 8	126
ค9 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 9	127
ค10 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 10	127
ค11 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 11	128
ค12 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 12	128
ค13 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 13	129
ค14 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 14	129

คำอธิบายสัญลักษณ์และคำย่อ

a	=	ค่าเอาต์พุต
b	=	ไบแอส
$E(k)$	=	ความคลาดเคลื่อนของชุดข้อมูลที่ k
J	=	คณณนิสมรณณะ
JNN_c	=	คณณนิสมรณณะของการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุม
JNN_m	=	คณณนิสมรณณะของการหาเอกลักษณ์ด้วยโครงข่ายประสาท
N	=	จำนวนชุดข้อมูลที่ใช้ในการฝึกโครงข่ายประสาทเทียม
n	=	ผลรวมอินพุต
P	=	ชุดของข้อมูลที่ใช้ในการฝึกโครงข่ายประสาทเทียม
$Ref(k)$	=	สัญญาณอ้างอิง
T	=	ข้อมูลเป้าหมาย
U	=	สัญญาณอินพุต
U_c	=	สัญญาณควบคุม
V	=	ค่าถ่วงน้ำหนักในชั้นซ่อน
V_{Bisas}	=	ค่าไบแอสในชั้นซ่อน
W	=	ค่าถ่วงน้ำหนักในชั้นเอาต์พุต
W_{Bisas}	=	ค่าไบแอสในชั้นเอาต์พุต
w	=	ค่าถ่วงน้ำหนัก
X	=	ข้อมูลอินพุต
Y	=	สัญญาณที่ออกจากชั้นเอาต์พุต
Z	=	สัญญาณที่ออกจากชั้นซ่อน
α	=	อัตราการเรียนรู้
δ_h	=	ค่าความคลาดเคลื่อนในชั้นซ่อน
δ_o	=	ค่าความคลาดเคลื่อนในชั้นเอาต์พุต

การควบคุมแรงบิดของเครื่องขันสกรูอัตโนมัติโดยเทคนิคโครงข่ายประสาทเทียม

Torque Control of Auto-Screwdriver Machine by Neural Network

คำนำ

อุตสาหกรรมฮาร์ดดิสก์ไดรฟ์ (Hard Disk Drive) ในปัจจุบันมีอัตราแข่งขันสูงมาก ถ้าบริษัทใดผลิตสินค้าออกมาได้มากกว่าและรวดเร็วกว่า บริษัทนั้นก็จะได้เปรียบทางการตลาดสูง เช่นเดียวกันหากสายการผลิตต้องหยุดชะงักลง ก็อาจจะทำให้เกิดการสูญเสียรายได้หลายล้านบาท ดังนั้นบริษัทผู้ผลิตจึงต้องทำให้สายการผลิตของตนเองมีประสิทธิภาพสูงสุด เพื่อที่จะได้ต่อสู้กับคู่แข่งได้ โดยสายการผลิตหลักของอุตสาหกรรมฮาร์ดดิสก์ไดรฟ์คือ การประกอบ (Assembly) และเครื่องมือที่สำคัญสำหรับการประกอบคือเครื่องขันสกรู (Screwdriver Machine)

ในอุตสาหกรรมฮาร์ดดิสก์ไดรฟ์การประกอบนั้นถือว่าเป็นสิ่งสำคัญมากส่วนหนึ่ง เพราะอุปกรณ์ที่สำคัญทุกตัวไม่ว่าจะเป็นหัวอ่านเขียนข้อมูล (Slider or Read/Write Head) หรือ แผ่นบันทึกข้อมูล (Media) จะมีการยึดด้วยการขันสกรู และทั้งหมดจะถูกบรรจุอยู่ในกล่องปกคลุม (Case) ทั้งหมด โดยการปิดกล่องปกคลุมนั้นจะใช้การขันสกรูเช่นเดียวกัน ดังนั้นการขันสกรูจึงสำคัญมาก ถ้าหากขันหลวมไปฝุ่นผงอาจจะเข้าไปในกล่องฮาร์ดดิสก์ได้ หรือถ้าขันสกรูแน่นเกินไปก็อาจจะทำให้ชิ้นงานเสียหายได้ ดังนั้นจำเป็นต้องมีการขันสกรูในการแรงบิดที่เหมาะสม ซึ่งการขันสกรูแต่ละจุดนั้นจะใช้แรงบิดไม่เท่ากัน เครื่องที่ใช้ในการขันสกรูจึงต้องมีความแม่นยำสูงเพื่อให้ชิ้นงานจะได้ไม่เกิดความเสียหายเนื่องจากการขันสกรู

ดังที่กล่าวมาแล้วนั้น การขันสกรูเพื่อประกอบฮาร์ดดิสก์เป็นกระบวนการสำคัญที่ต้องการความแม่นยำมาก ยิ่งฮาร์ดดิสก์ที่มีขนาดเล็กลงยิ่งต้องระมัดระวังมากขึ้น เนื่องจากในฮาร์ดดิสก์ตัวเล็กต้องใช้แรงบิดในการขันสกรูน้อยกว่าในฮาร์ดดิสก์ตัวใหญ่ทำให้ยากต่อการควบคุม จากการที่ผู้วิจัยได้ศึกษาระบบการทำงานของเครื่องขันสกรู พบว่าระบบควบคุมของเครื่องขันสกรูหลายๆ เครื่องส่วนใหญ่เป็นระบบควบคุมแบบเปิดหรือไม่มีระบบการควบคุมแบบป้อนกลับของแรงบิด ซึ่งอาจส่งผลให้เครื่องทำงานได้ไม่ตรงกับความต้องการได้ ทำให้เกิดการศึกษาเพื่อพัฒนาระบบควบคุมแบบปิดของการขันสกรูซึ่งมีหลายวิธี โดยจุดประสงค์ที่สำคัญคือ เพื่อควบคุมแรงบิดให้แม่นยำที่สุด

ในการออกแบบระบบควบคุมแรงบิดนั้น ได้ทำการออกแบบสัญญาณควบคุมเพื่อบังคับการทำงานของมอเตอร์แกน R ซึ่งเป็นมอเตอร์ที่ใช้ในการขันสกรู โดยใช้ระบบป้อนกลับควบคุมกับอุปกรณ์วัดแรงบิด (Transducer Torque) และใช้เทคนิคโครงข่ายประสาทเทียมเข้ามาออกแบบระบบควบคุมด้วย เพื่อให้ได้ผลตอบสนองออกมาแม่นยำขึ้น



วัตถุประสงค์

1. พัฒนาระบบการขันสกรูจากระบบควบคุมแบบเปิดให้เป็นระบบควบคุมแบบปิด
2. ออกแบบระบบควบคุมแบบปิดโดยเทคนิคโครงข่ายประสาทเทียม

ขอบเขตของงานวิจัย

1. ควบคุมแรงบิดในกระบวนการขันสกรูฮาร์ดดิสก์ไฟฟ้าให้อยู่ในช่วง 0.2 ถึง 1 Nm
2. ความเร็วของมอเตอร์ใช้ในกระบวนการขันสกรูฮาร์ดดิสก์ไฟฟ้าให้อยู่ในช่วงความเร็วไม่เกิน 2400 rpm

ประโยชน์ที่คาดว่าจะได้รับ

1. ระบบควบคุมแบบปิดสำหรับเครื่องขันสกรู
2. เครื่องขันสกรูที่มีประสิทธิภาพการทำงานสูงขึ้น
3. ลดอัตราการเสียหายของชิ้นงาน ที่เกิดจากการขันสกรูดั้วแรงบิดที่ไม่เหมาะสม
4. สามารถนำหลักการที่ได้จากการวิจัยนี้ไปประยุกต์ใช้กับเครื่องขันสกรูประเภทอื่นๆได้

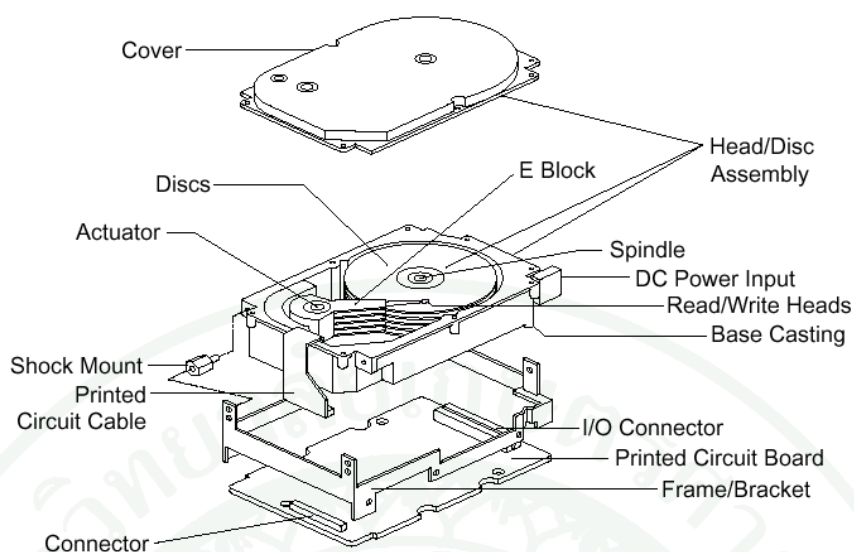
การตรวจเอกสาร

1. ฮาร์ดดิสก์ไดรฟ์ (Hard Disk Drive)

ฮาร์ดดิสก์ไดรฟ์ (Hard Disk Drive) หรือ ฮาร์ดดิสก์ (Hard Disk) จัดเป็นอุปกรณ์สื่อบันทึกข้อมูลประเภทหนึ่ง (Storage Device) ที่นิยมใช้เก็บข้อมูลต่างๆ ภายในเครื่องคอมพิวเตอร์ เนื่องจากข้อมูลหรือโปรแกรมต่างๆ ในปัจจุบันมีขนาดใหญ่ไม่สามารถจัดเก็บลงดิสเกตต์ได้จึงจำเป็นต้องเก็บลงฮาร์ดดิสก์ซึ่งมีความจุที่สูงกว่า เมื่อมีการเรียกโปรแกรมหรือข้อมูลต่างๆ ก็จะไม่โหลดตัวเองเข้าแรม(RAM) และเมื่อมีการแก้ไขหรือการอัปเดตข้อมูลใหม่ก็จะเก็บข้อมูลใหม่นั้นลงในฮาร์ดดิสก์อีกครั้ง ดังนั้นฮาร์ดดิสก์จึงเป็นอุปกรณ์ที่มีความสำคัญอย่างมากภายในเครื่องคอมพิวเตอร์

1.1 ส่วนประกอบและโครงสร้างของฮาร์ดดิสก์ไดรฟ์

ภายในตัวฮาร์ดดิสก์นั้นจะประกอบไปด้วยส่วนต่างๆ มากมาย โดยส่วนต่างๆ เหล่านี้จะถูกปกปิดไว้ด้วยเคส (Case) ที่เป็นตัวถังโลหะ ซึ่งปิดผนึกแน่นเป็นสุญญากาศไม่สามารถให้ฝุ่นผงหรือสิ่งใดหลุดรอดเข้าไปได้ ส่วนโครงสร้างของฮาร์ดดิสก์เป็นลักษณะกลไก หรือ Mechanical การอ่านหรือเขียนข้อมูลนั้นจะต้องอาศัยการเคลื่อนไหวของชิ้นส่วนต่างๆ ซึ่งถือว่าเป็นการทำงานในรูปแบบอิเล็กทรอนิกส์ โดยมีส่วนประกอบที่สำคัญแสดงในภาพที่ 1



ภาพที่ 1 ส่วนประกอบหลัก ภายในฮาร์ดดิสก์ไครฟ์

ที่มา: Seagate Technology (2000)

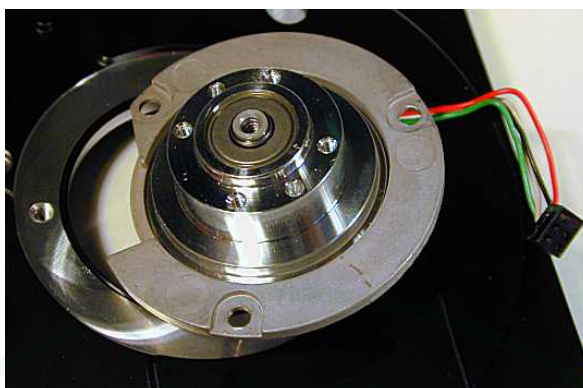
1.1.1 ดิสก์บันทึกข้อมูล มีชื่อเรียกเพลตเตอร์ (Platter) หรือแผ่นมีเดีย (Media) เป็นแผ่นจานกลมที่มีลักษณะแข็งและแบนราบ ซึ่งทำจากแก้วหรือโลหะผสมอลูมิเนียม พื้นผิวทั้งสองด้านของแผ่นมีเดียแต่ละแผ่นจะถูกเคลือบไว้ด้วยสารที่มีคุณสมบัติทางแม่เหล็ก เนื่องจากการบันทึกข้อมูลจะใช้หลักการเหนี่ยวนำของสนามแม่เหล็ก และการบันทึกข้อมูลนั้นจะสามารถบันทึกได้ทั้งสองด้านของแผ่นมีเดีย โดยมีเส้นผ่านศูนย์กลางอยู่ระหว่าง 2 ถึง 5.25 นิ้ว ซึ่งจานมีเดียแต่ละแผ่นจะวางซ้อนกันอยู่ มีเดียเหล่านี้จะถูกเจาะรูที่กึ่งกลางและยึดติดเข้าด้วยกันกับแกนหมุนที่เรียกว่า สปินเดิล (Spindle) โดยมีมอเตอร์ (Spindle Motor) เป็นตัวขับเคลื่อนให้แผ่นมีเดียทุกแผ่นหมุนไปพร้อมกันและด้วยความเร็วที่เท่ากัน มีเดียแต่ละแผ่นที่วางซ้อนกันอยู่นั้น จะมีหัวอ่านเขียนข้อมูลประกบทั้งด้านบนและล่างของแต่ละแผ่น หัวอ่านเขียนข้อมูลของฮาร์ดดิสก์จะลอยอยู่เหนือจากแผ่นมีเดียเพียง 0.5 ไมครอน



ภาพที่ 2 แผ่นเพลดเตอร์ หรือแผ่นมีเดียที่ใช้บันทึกข้อมูล

ที่มา: รีเพอร์โน้ตบุ๊กดอตเน็ต (2009)

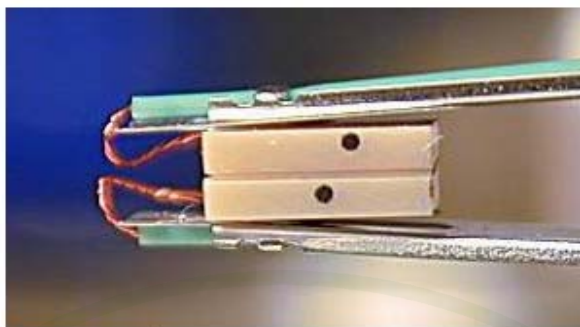
1.1.2 สปินเดิล (Spindle) เป็นอุปกรณ์ที่ทำหน้าที่หมุนแผ่นมีเดียด้วยความเร็วคงที่ค่าหนึ่ง โดยจะมีมอเตอร์ (Spindle Motor) เป็นตัวขับเคลื่อนให้แผ่นมีเดียทุกแผ่นหมุนไปพร้อมๆ กัน เพื่อให้ผิวของแผ่นมีเดียเคลื่อนที่ผ่านหัวอ่านเขียนข้อมูล ซึ่งทำหน้าที่อ่านข้อมูลที่บันทึกอยู่บนแผ่นมีเดียความเร็วในการหมุนแผ่นนี้มีผลต่อความเร็วในการเข้าถึงข้อมูลและความเร็วในการอ่านข้อมูลด้วย ยิ่งถ้ามีความเร็วเพิ่มมากขึ้น เวลาที่ใช้ในการเข้าถึงข้อมูลก็จะน้อยลง และในขณะเดียวกันแผ่นมีเดียก็จะเคลื่อนที่ผ่านหัวอ่านเขียนข้อมูลได้เร็วขึ้นด้วย จึงทำให้สามารถอ่านข้อมูลได้เร็วขึ้นเพราะส่วนของข้อมูลที่ต้องการจะถูกหมุนมาอยู่ภายใต้หัวอ่านเขียนข้อมูลนี้อย่างรวดเร็ว



ภาพที่ 3 สปินเดิลมอเตอร์

ที่มา: เดอะพีซีไกด์ (2547)

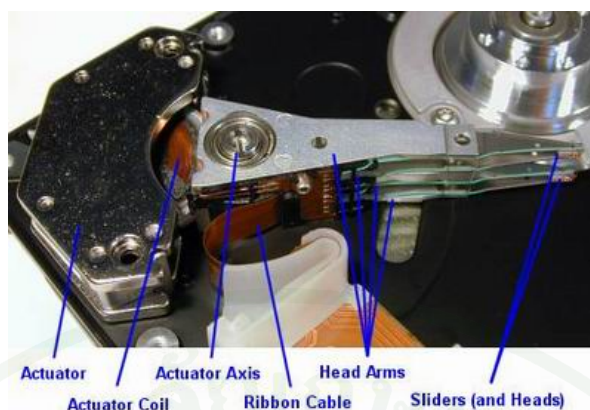
1.1.3 หัวอ่านเขียนข้อมูล (Slider or Read/Write Head) ทำหน้าที่รับผิดชอบในการอ่านและเขียนข้อมูลจากแผ่นมีเดีย ลักษณะทางกายภาพก็จะติดอยู่กับปลายแขนของหัวอ่านเขียนข้อมูล (Actuator arms) หัวอ่านเขียนข้อมูลนี้จะมียู้งทั้งด้านบนและล่างของแผ่นมีเดียแต่ละแผ่น หมายความว่าแผ่นมีเดียแต่ละแผ่นนั้น จะมีหัวอ่านเขียนข้อมูลอยู่สองหัวอ่านด้วยกัน เมื่อตัวไครฟ์เริ่มทำงานตัวสปินเดิลมอเตอร์ (Spindle Motor) จะทำหน้าที่หมุนแผ่นมีเดียด้วยความเร็วที่กำหนดไว้ของฮาร์ดดิสก์ การหมุนด้วยความเร็วนี้เองทำให้เกิดการไหลเวียนของอากาศหรือที่เรียกว่า air flow ซึ่งจะช่วยยกส่วนแขนของหัวอ่านเขียนข้อมูลให้สูงขึ้นจากระดับเดิมเหนือแผ่นมีเดียเล็กน้อย ระยะห่างระหว่างแผ่นมีเดียกับหัวอ่านเขียนข้อมูลเพียง 0.5 ไมครอนเท่านั้น ประโยชน์ของการที่หัวอ่านเขียนข้อมูลลอยอยู่เหนือแผ่นมีเดียนี้จำเป็นมากต่อการทำงานของฮาร์ดดิสก์ เนื่องจากแผ่นมีเดียนั้นจะถูกหมุนด้วยความเร็วที่สูงมาก ถ้าหากหัวอ่านเขียนข้อมูลแตะกับแผ่นมีเดียแล้ว ย่อมจะทำให้เกิดรอยขีดข่วนอย่างถาวร ซึ่งทำให้ฮาร์ดดิสก์เกิดความเสียหายและเกิดความผิดพลาดในการอ่านหรือเขียนข้อมูลได้ หรืออาจจะไม่สามารถใช้งานได้อีกต่อไป



ภาพที่ 4 แสดงลักษณะหัวอ่านเขียนข้อมูล

ที่มา: เดอะพีซีไกด์ (2547)

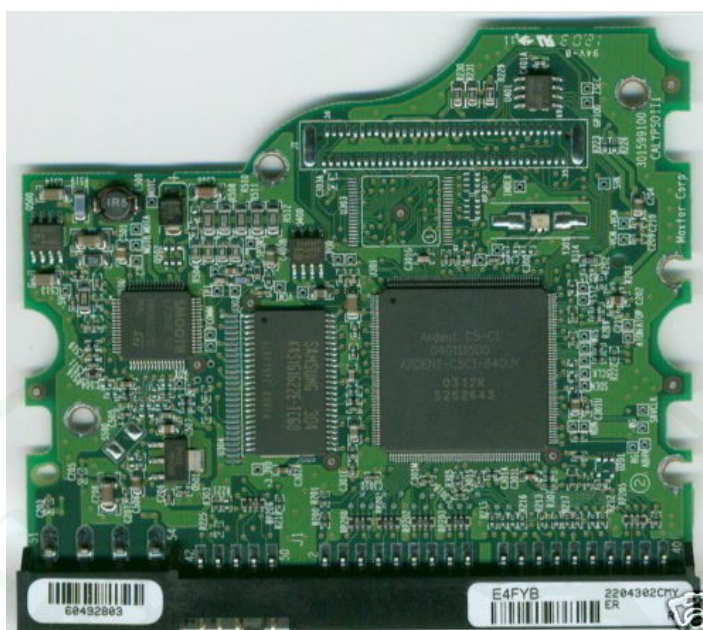
1.1.4 แอคชูเอเตอร์ (Actuator) เป็นส่วนประกอบที่ทำหน้าที่เคลื่อนแขนของหัวอ่านเองซึ่งถูกเรียกว่า แขนของหัวอ่านเขียนข้อมูล (Actuator Arms or Head Arms) อุปกรณ์ชุดนี้ประกอบด้วยแม่เหล็กแรงสูง ขดลวดทองแดง และก้านอลูมิเนียม ซึ่งตรงปลายของมันจะมีหัวอ่านเขียนข้อมูลติดอยู่ ขณะที่ทำงานตัวควบคุม (Controller) จะทำหน้าที่ควบคุมให้แขนของหัวอ่านเขียนข้อมูล เลื่อนหัวอ่านไปยังตำแหน่งที่ต้องการ ตัวแอคชูเอเตอร์นี้เป็นส่วนหนึ่งที่สำคัญในฮาร์ดดิสก์และเป็นตัวบ่งบอกถึงความเร็วในการทำงานของตัวไดรฟ์และจำนวนบิตที่สามารถบรรจุไว้บนตัวแผ่นแพตเตอร์ ตัวแอคชูเอเตอร์จะต้องเคลื่อนหัวอ่านเขียนข้อมูลไปในแนวตรงอย่างแม่นยำ ไม่เช่นนั้นแล้วจะไม่สามารถอ่านหรือเขียนข้อมูลได้อย่างถูกต้อง นอกจากนี้แอคชูเอเตอร์จะเกี่ยวข้องโดยตรงกับความก้าวหน้าหรือการพัฒนาในส่วนความเร็ว (Speed) และความจุ (Capacity) ของฮาร์ดดิสก์ เหตุเพราะว่ายิ่งข้อมูลถูกบันทึกไว้ใกล้กันหรือชิดกันมากเท่าใดบนแผ่นแพตเตอร์ ส่วนของแอคชูเอเตอร์ก็จะต้องสามารถเคลื่อนแขนหัวอ่านเขียนข้อมูลเข้าหาตำแหน่งที่ต้องการได้ยากขึ้นเท่านั้น ถ้าเราไม่ได้พัฒนาตัวแอคชูเอเตอร์นี้ให้ทำงานอย่างแม่นยำถูกต้องก็จะเป็นการถ่วงพัฒนาการของเทคโนโลยีการผลิตฮาร์ดดิสก์ให้ด้อยลงตามไปด้วย



ภาพที่ 5 แสดงส่วนประกอบต่างๆ ของแอกชูเอเตอร์ (Actuator)

ที่มา: เดอะพีซีไกด์ (2547)

1.1.5 แผงวงจร (Circuit or Logic Board) ฮาร์ดดิสก์แต่ละตัวจะมีแผงวงจร ซึ่งจะถูกยึดติดกับภายนอกของตัวฮาร์ดดิสก์โดยจะมีชิพไอซีที่เรียกว่า DSP (Digital Signal Processors) จะควบคุมการส่งผ่านข้อมูล, Cache Memory จะเก็บข้อมูลส่วนที่เรียกใช้บ่อยๆ รวมทั้งส่วนของการควบคุม สปีดเคิล แอพทิวเลเตอร์ การปฏิบัติงานของหัวอ่านเขียนข้อมูล และ Power Management แผงวงจรนี้จะรับคำสั่งผ่านทางส่วนควบคุมการทำงานของฮาร์ดดิสก์ (Harddisk's Controller) ซึ่งถูกควบคุมโดยระบบปฏิบัติการอีกทีหนึ่ง เมื่อระบบต้องการข้อมูลที่เก็บบันทึกอยู่ในฮาร์ดดิสก์ ตัวแผงวงจรนี้จะรับคำสั่งดำเนินการตามคำสั่งโดยเคลื่อนและควบคุมส่วนประกอบต่างๆ เพื่อให้ได้รับข้อมูลที่ต้องการออกมา



ภาพที่ 6 แสดงแผงวงจรของฮาร์ดดิสก์

ที่มา: ไอออฟฟอร์คอตคอม (2549)

1.1.6 กล่องใส่ฮาร์ดดิสก์ หรือเคส (Case) นี้สำหรับบรรจุแผ่นดิสก์ และกลไกในการหมุน การอ่าน ของฮาร์ดดิสก์ ผู้ผลิตแต่ละรายจะผลิตเคสไม่เหมือนกัน ปกติเคสนี้จะถูกปิดผนึก (Seal) อย่างแน่นหนา เพื่อป้องกันฝุ่นเข้าไปในเคส

2. ระบบควบคุม (Control System)

การควบคุมในระบบกายภาพใดๆ คือ การบังคับให้ระบบนั้นทำงานในลักษณะที่จะนำมาซึ่งผลงาน (Output) ที่มีคุณสมบัติสอดคล้อง หรือเป็นไปตามเป้าหมาย (Input) ซึ่งแสดงได้โดยการเขียนหัวลูกศรออกและเข้าสู่ระบบลำดับ ดังแสดงในภาพที่ 7 แต่ในระบบควบคุมทั่วไป Output และ Input อาจเป็นปริมาณหรือสัญญาณที่เหมือนกัน หรือไม่เหมือนกันก็ได้



ภาพที่ 7 OUTPUT และ INPUT ของระบบควบคุม

การศึกษาระบบควบคุม เป็นการศึกษาถึงการตอบสนองของระบบ (System Response) ซึ่งได้แก่ ลักษณะของ Output หรือ Input ที่เข้าสู่ระบบ การตอบสนองของระบบต่อ Input ที่เข้าสู่ระบบ สามารถแบ่งออกได้เป็น การตอบสนองส่วนที่เปลี่ยนแปลงตามเวลา หรือที่เรียกว่า Transient Response กับ การตอบสนองส่วนที่ไม่เปลี่ยนแปลงตามเวลา หรือที่เรียกว่า Steady State Response ซึ่งโดยทั่วไป การศึกษาระบบควบคุมจะเน้นไปที่ การตอบสนองส่วนที่เปลี่ยนแปลงตามเวลา ทำให้การวิเคราะห์ วิจัย ระบบควบคุมเสมือนเป็นการวิเคราะห์ระบบพลวัต (Dynamics System) และในบางครั้งจะพบว่า ทฤษฎีหรือหลักเกณฑ์ของระบบควบคุมบางอย่าง สามารถนำไปประยุกต์ใช้กับระบบพลวัตอื่นๆ เช่น ระบบสั่นสะเทือนหรือ ในทางกลับกันได้เป็นอย่างดี

2.1 คุณภาพของระบบควบคุม

ในการเปรียบเทียบคุณภาพของระบบควบคุมที่เหมือนกัน 2 ระบบว่าระบบไหนดีกว่ากัน จะต้องดูที่การตอบสนอง (Response) ของแต่ละระบบแล้วนำมาเปรียบเทียบกัน ระบบที่ให้การตอบสนองที่แม่นยำ ย่อมดีกว่าระบบที่ให้การตอบสนองที่คลาดเคลื่อน, ระบบที่ให้การตอบสนองที่ฉับไว ย่อมดีกว่าระบบที่ให้การตอบสนองอย่างเฉื่อยชา (Sluggish) และระบบที่ให้การตอบสนองที่เสถียร (Stable) ย่อมดีกว่าระบบที่ให้การตอบสนองที่ไม่เสถียร (Unstable Response) หรือคุมไม่ได้ จึงกล่าวได้ว่า คุณภาพของระบบควบคุม หมายถึง

- ความแม่นยำ (Accuracy)
- ความรวดเร็ว (Speed)
- ความเสถียร (Stability)

ของการตอบสนอง ของระบบนั้นๆ

ระบบควบคุมที่ขาดความแม่นยำย่อมนำไปใช้ประโยชน์ไม่ได้ และถือว่าเป็นระบบการควบคุมที่สร้างขึ้น หรือทำขึ้นโดยผิดวัตถุประสงค์ของการควบคุม และแน่นอนความแม่นยำ

ในที่นี้ รวมถึงความสม่ำเสมอคงเส้นคงวาในการตอบสนองด้วย ความรวดเร็วในการตอบสนองก็เช่นกัน ระบบที่มีการตอบสนองที่แม่นยำไม่มีความคลาดเคลื่อนเลย แต่ใช้เวลานานมากกว่าจะถึง Steady State ก็ย่อมเป็นระบบควบคุมที่นำไปใช้งานไม่ได้ ความเสถียรของการตอบสนองของระบบควบคุมเป็นสิ่งสำคัญที่จะชี้ให้เห็นถึงคุณภาพของระบบควบคุมนั้นๆ ระบบควบคุมที่มีความรวดเร็วและแม่นยำในการตอบสนอง แต่ในขณะหรือกับบางลักษณะของ Input ไม่สามารถให้การตอบสนองที่เสถียร ระบบการควบคุมนั้นก็ใช้งานไม่ได้และชำรุดไป

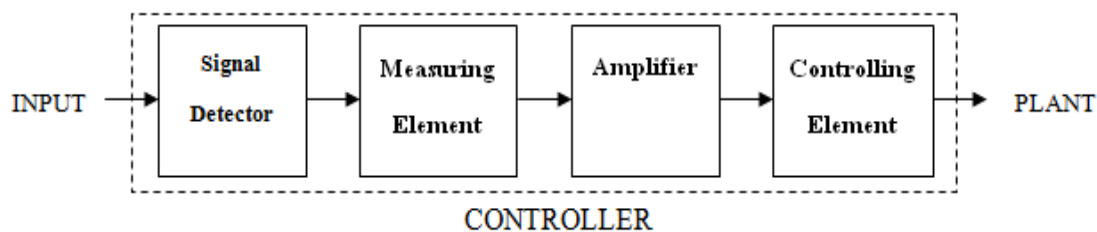
จึงเห็นได้ว่า ความแม่นยำ ความรวดเร็วและความเสถียร นอกจากจะเป็นสิ่งที่ชี้ให้เห็นถึงคุณภาพของระบบควบคุมแล้ว ยังเป็นสิ่งจำเป็นสำหรับระบบควบคุมทุกระบบ จะขาดอันใดอันหนึ่งไปไม่ได้ ในระบบควบคุมหนึ่งๆ ผู้ออกแบบ หรือผู้สร้างระบบควบคุมนั้นๆ จะต้องคำนึงถึงทั้ง 3 องค์ประกอบ คือ พยายามให้ระบบควบคุมที่ทำงาน หรือให้การตอบสนองที่แม่นยำที่สุด ที่เร็วที่สุด และที่เสถียรที่สุด ซึ่งในทางปฏิบัติทำได้ยาก เพราะขณะที่พยายามให้ระบบควบคุมมีการตอบสนองที่รวดเร็ว โดยทั่วไปจะทำให้ระบบเสียความเสถียรไป หรือขณะที่พยายามให้ระบบมีการตอบสนองที่แม่นยำขึ้น โดยทั่วไปก็จะมีผลกระทบทำให้การตอบสนองล่าช้าลง

2.2 ส่วนประกอบของระบบควบคุม

ระบบควบคุม ประกอบด้วยส่วนประกอบใหญ่ๆ 2 ส่วน คือ ส่วนที่เป็นตัวควบคุม (Controller) และส่วนที่เป็นต้นกำลัง (Plant)

2.2.1 ส่วนที่เป็นตัวควบคุม (Controller)

ตัวควบคุม หมายถึง ส่วนซึ่งทำหน้าที่แปรสัญญาณจากภายนอก ส่งเข้าสู่ระบบ หรือ Input ให้เป็นสัญญาณอื่นที่เหมาะสมและมีกำลังพอ เพื่อไปบังคับให้ส่วนที่เป็นต้นกำลังทำงาน เพื่อให้ระบบควบคุมทั้งระบบทำงาน โดยทั่วไปส่วนที่เป็นตัวควบคุมประกอบด้วย ตัวรับสัญญาณ หรือ Signal Detector หรือ Receiver ตัววัดระดับสัญญาณ หรือ Measuring Device และตัวขยายสัญญาณ หรือ Signal Amplifier และตัวควบคุม หรือ Controlling Element ซึ่งทำหน้าที่เปลี่ยนรูปหรือ ลักษณะของสัญญาณ ดังภาพที่ 8



ภาพที่ 8 ส่วนที่เป็นตัวควบคุมและส่วนประกอบต่างๆ

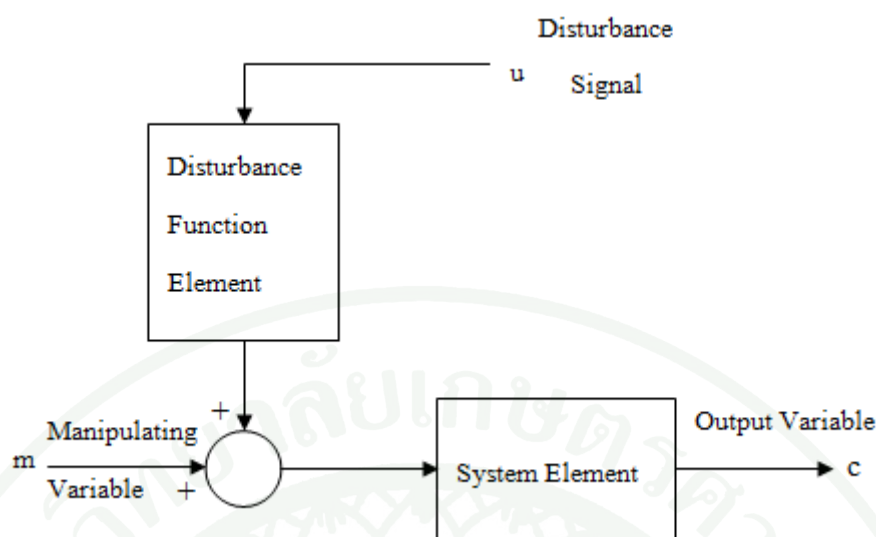
2.2.2 ส่วนที่เป็นต้นกำลัง (Plant)

ต้นกำลัง คือ ส่วนของระบบควบคุมที่รับสัญญาณ Manipulating Variable (m) จากส่วนควบคุม หรือ Controller แล้วทำงานให้ Output หรือ Controller Variable ซึ่งใช้สัญลักษณ์ c ซึ่งอาจเขียนแสดงได้ดังภาพที่ 9



ภาพที่ 9 สัญญาณเข้าและออกของ Plant

เนื่องจากในระบบควบคุมทั่วไป สัญญาณที่เข้าสู่ตัวทำงาน ซึ่งเรียกว่า System Element มักจะประกอบด้วยสัญญาณอื่น เนื่องมาจากการรบกวนจากภายนอกซึ่งเรียกว่า External Disturbance ส่วน Plant จึงประกอบด้วย ตัวรวมสัญญาณ หรือ Signal Summing Element ซึ่งทำหน้าที่รวมสัญญาณจากภายนอกเข้ากับ Manipulating Variable แล้วจึงส่งต่อไปยัง System Element ดังแสดงในภาพที่ 10



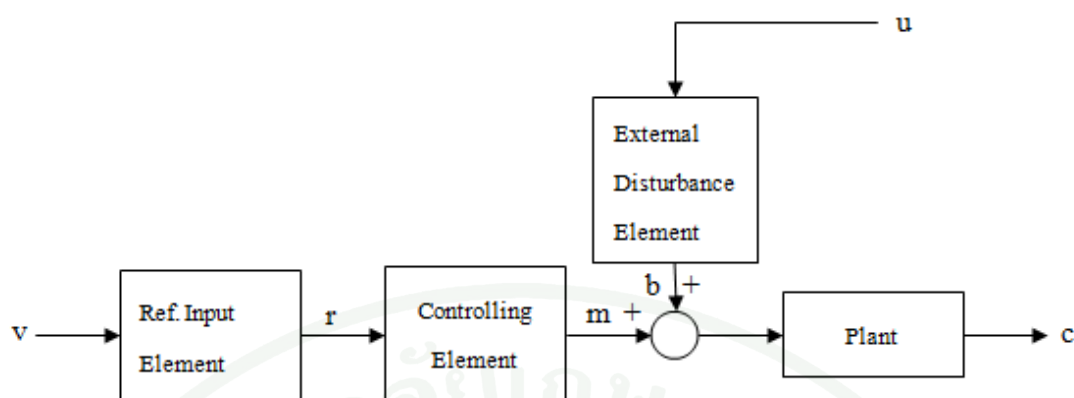
ภาพที่ 10 Plant ของระบบควบคุมที่มีการรบกวนจากภายนอก

จะเห็นว่า ข้อแตกต่างของส่วนคอนโทรลเลอร์กับส่วนแพลนท์ ก็คือ คอนโทรลเลอร์ทำหน้าที่ควบคุมการทำงานของทั้งระบบ ให้เป็นไปตามเป้าหมายของระบบควบคุม แต่เอาต์พุตของส่วนนี้ คือ Manipulating Variable ไม่สามารถทำงานให้ระบบได้ จึงต้องอาศัยแพลนท์อีกต่อหนึ่ง สำหรับแพลนท์จะทำหน้าที่ผลิตกำลังให้เป็นไปตามระดับความแรงของสัญญาณ Manipulating Variable จากส่วนคอนโทรลเลอร์ เพื่อให้ได้เอาต์พุตของระบบ กล่าวคือ แพลนท์จะทำงานในลักษณะแปรรูป หรือส่งผ่านพลังงาน

2.3 ประเภทของระบบควบคุม

การควบคุมระบบสามารถแบ่งออกได้เป็น 2 ประเภท คือ ประเภทวงจรเปิด หรือประเภทที่ไม่มีการป้อนกลับ กับประเภทวงจรปิด หรือประเภทที่มีการป้อนกลับ

2.3.1 ประเภทวงจรเปิด หรือประเภทที่ไม่มีการป้อนกลับ (Open-loop Control System)

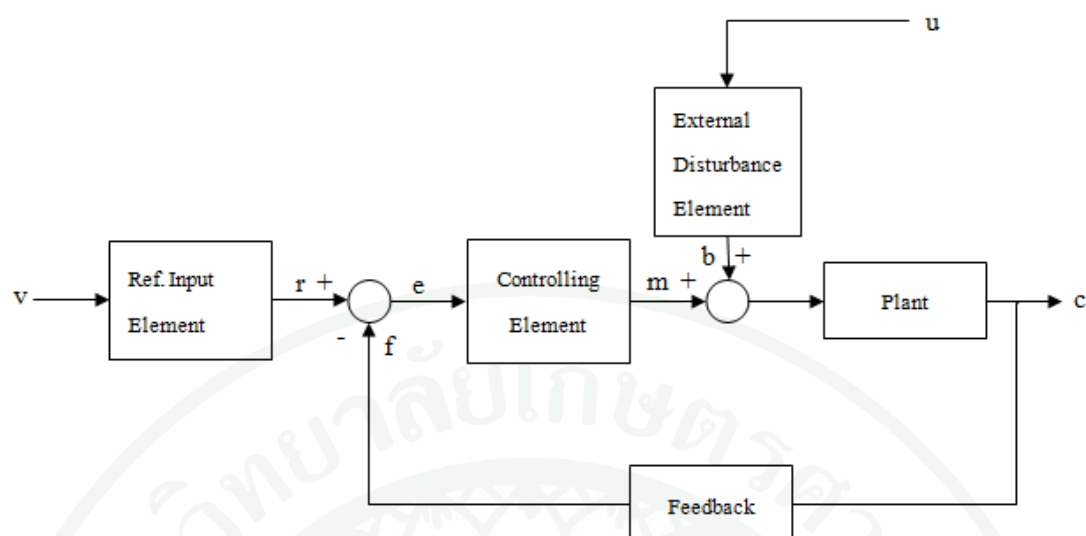


ภาพที่ 11 ระบบควบคุมแบบวงจรเปิด

ระบบควบคุมประเภทนี้ เป็นการควบคุมที่ไม่มีการเปรียบเทียบสภาพของเอาต์พุตกับอินพุตของระบบ โดยทั่วไปแพลนท์จะทำงานที่อัตราคงที่ ข้อสังเกตที่สำคัญของระบบควบคุมประเภทนี้ก็คือ ความคลาดเคลื่อนของเอาต์พุตระบบ เพราะในทางปฏิบัติ ระบบจะถูกรบกวนจากภายนอกเสมอ ทำให้ส่วนแพลนท์ไม่สามารถทำงานได้ตามที่คอนโทรลเลอร์สั่งมา แต่จะเอาการรบกวนเข้ามาด้วย และไม่มีการแก้ไขกลับไป (Feedback) ที่ส่วนคอนโทรลเลอร์อีกแต่อย่างใด

2.3.2 ประเภทวงจรรปิด หรือประเภทที่มีการป้อนกลับ (Close-loop Control System)

ระบบควบคุมประเภทนี้ เป็นระบบการควบคุมที่เอาต์พุตของระบบหรือ Controller Variable (c) ถูกป้อนกลับมายังส่วนคอนโทรลเลอร์เพื่อมาเปรียบเทียบกับ Reference Variable (r) โดยอาศัยตัวเปรียบเทียบค่าตลอดเวลาในช่วง Transient Response ของระบบควบคุมประเภทนี้ Error ซึ่งมีสัญลักษณ์ e จะเปลี่ยนค่าตลอดเวลา ถึงแม้ Reference Variable จะไม่เปลี่ยนค่าก็ตาม เนื่องจาก Feedback Variable เปลี่ยนค่าตลอดเวลา เพราะ Controller Variable เปลี่ยนค่าตลอดเวลา ฉะนั้นส่วนที่เป็นแพลนท์ของระบบการควบคุมประเภทนี้ โดนทั่วไปแล้วจะทำงานด้วยอัตราที่ไม่คงที่



ภาพที่ 12 ระบบควบคุมแบบวงจรมีป้อนกลับ

3. โครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทเทียม เป็นแบบจำลองทางคณิตศาสตร์ที่ได้รับการพัฒนาโดยอาศัยหลักการการทำงานของระบบประสาททางชีววิทยา สามารถเปรียบเทียบขั้นตอนต่างๆ ระหว่าง เซลล์ประสาททางชีววิทยา กับ เซลล์ประสาทเทียมได้โดย โครงสร้างของเซลล์ประสาทมีการเชื่อมต่อกันอย่างมากมายด้วยการส่งสัญญาณไฟฟ้าทางเคมี (Electrochemical) จากเซลล์หนึ่งไปอีกเซลล์หนึ่งผ่าน เดนไดรต์ ซึ่งเปรียบเสมือนเป็นสัญญาณอินพุต (Input) ผ่านไซแนปส์ ซึ่งเปรียบเสมือนเป็นค่าถ่วงน้ำหนัก (Weight) จากนั้นสัญญาณจะถูกส่งเข้าไปกระบวนการการประมวลผล ซึ่งประกอบไปด้วยหน่วยประมวลผล (Processing elements) ที่เรียกว่า นิวรอน (Neurons) ยูนิตส์ (Units) เซลล์ (Cells) หรือ โหนด (Nodes) ภายในตัวเซลล์ หลังจากนั้นจะส่งสัญญาณผ่านแอกซอน ซึ่งเปรียบเสมือนเป็นสัญญาณเอาต์พุต (Output) ไปยังเซลล์อื่นๆ ต่อไปในลักษณะการเชื่อมต่อแบบเดียวกันนี้เรื่อยๆ ไป ซึ่งถ้าเกิดการไซแนปส์เป็นแบบการกระตุ้น (Excitatory Synapse) มีผลทำให้สัญญาณมีความถี่สูงขึ้น แต่ถ้าเกิดการไซแนปส์เป็นแบบยับยั้ง (Inhibitory) มีผลทำให้สัญญาณมีความถี่ต่ำลง การเชื่อมต่อนี้เองจะมีการปรับค่าถ่วงน้ำหนักของแต่ละจุดของการเชื่อมต่อให้ได้ค่าน้ำหนักที่เหมาะสมในการนำโครงข่ายประสาทเทียมไปประยุกต์ เพื่อให้สอดคล้องในการแก้ไขปัญหาต่างๆ เช่น การเรียนรู้และจดจำรูปแบบ (Pattern Recognition) การประมวลผลสัญญาณ (Signal Processing) การควบคุมการทำงานของหุ่นยนต์ (Control Robot) และการพยากรณ์ (Forecasting) เป็นต้น

3.1 โครงสร้างของโครงข่ายประสาทเทียม

รูปแบบของโครงข่ายประสาทเทียมจะประกอบด้วย Node ในแต่ละชั้น Layer และการเชื่อมต่อกันภายในระหว่างชั้น Layer ดังนี้

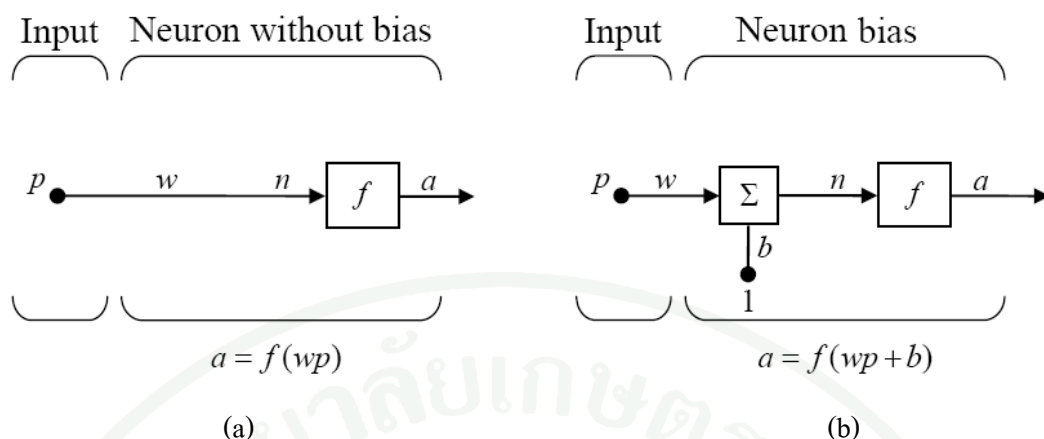
3.1.1 ชั้นแรกจะเป็นชั้น Input layer จะมีเพียงชั้นเดียวจำนวน Node จะมีเท่ากับจำนวนของข้อมูล Input Data การคำนวณในชั้น Input Node จะเป็นเพียงการแปลงค่าข้อมูลให้มีค่าระหว่าง 0 ถึง 1 เนื่องจากข้อมูลดิบจะเป็นตัวเลขที่มีค่ามาก และหน่วยของข้อมูลมีรูปแบบที่ไม่เหมือนกันจึงต้องแปลงให้อยู่ในรูปแบบและช่วงของข้อมูลเดียวกัน

3.1.2 ชั้น Hidden layer หรือชั้นซ่อน โดยทั่วไปจำนวนชั้นและจำนวน Node จะเป็นเท่าใดไม่มีข้อจำกัดแน่นอนตายตัว แต่ถ้ามีจำนวนชั้นและจำนวน Node มากๆ จะทำให้ช้ามากในการคำนวณ (Overfitting) และผลที่ได้ดีหรือไม่ดีไม่อาจจะทราบได้ และ ถ้ามีจำนวนชั้นและจำนวน Node น้อยเกินไป จะทำให้ผลการคำนวณค่าความคลาดเคลื่อนไม่ลู่เข้าจุดต่ำสุด (Convergent) และผลลัพธ์ที่ได้จะไม่ถูกต้อง ต้องทำการทดลองเปรียบเทียบกันระหว่าง Neural Network ที่มีจำนวนชั้น Hidden layer ต่างๆ กัน และจำนวน Node ในชั้น Hidden layer นั้นๆ ว่ามีจำนวนเท่าไรจะทำให้การคำนวณได้ผลดีกว่ากัน

3.1.3 ชั้น Output layer หรือชั้นเอาต์พุต จะมีชั้นเดียวและจะมีจำนวน Node เท่ากับจำนวน Output ที่ต้องการ

3.2 แบบจำลองของนิวรอน (Neuron Model)

3.2.1 Single-Input Neuron เป็นแบบจำลองนิวรอนอย่างง่าย ๆ มีค่าอินพุต (p) เพียงค่าเดียวเท่านั้น ซึ่งค่าอินพุตในกรณีนี้จะเป็นปริมาณสเกลาร์ จากภาพที่ 13 (a) แสดงแบบจำลองของนิวรอนอย่างง่ายในกรณีที่ไม่มีค่าไบแอส (b) ส่วนภาพที่ 13 (b) แสดงแบบจำลองของนิวรอนอย่างง่ายในกรณีที่มีค่าไบแอส



ภาพที่ 13 (a) นิวรอนที่มีอินพุตเพียงอินพุตเดียว โดยไม่นำค่าไบแอสมาคิด

(b) นิวรอนที่มีอินพุตเพียงอินพุตเดียว โดยนำค่าไบแอสมาคิด

ที่มา: Howard *et al.* (1996)

จากภาพที่ 13 จะเห็นได้ว่ามีค่าอินพุตถูกส่งผ่าน นำไปคูณด้วยค่าถ่วงน้ำหนัก (w) เป็นปริมาณสเกลาร์ กำหนดอยู่ในแต่ละการเชื่อมต่อ ผลคูณที่ได้อยู่ในรูป wp นำค่าที่ได้นี้ ไปรวมกับค่าไบแอสจะได้ผลรวมกันของค่าที่ได้เรียกว่า ค่า Net Input (n) จากนั้นถูกส่งผ่านไปยังฟังก์ชันการถ่ายโอนหรือฟังก์ชันการกระตุ้น (Transfer or Activation Function) ผลลัพธ์ที่ได้ออกมาเป็นค่าเอาต์พุต (a) เป็นปริมาณสเกลาร์ จากภาพสามารถเขียนสมการของค่าอินพุตและเอาต์พุตได้ดังนี้

$$a = f(wp + b) \quad (1)$$

$$n = wp + b \quad (2)$$

$$\therefore a = f(n) \quad (3)$$

ค่าไบแอสมีลักษณะเหมือนกับค่าถ่วงน้ำหนัก แต่จะกำหนดให้มีค่าเป็น 1 ในตอนเริ่มต้นหรือไม่ต้องการกำหนดให้มีค่าไบแอสก็ได้เช่นกัน ค่าถ่วงน้ำหนักและค่าไบแอสจะถูกปรับให้มีความสัมพันธ์กับฟังก์ชันการถ่ายโอนที่เลือกใช้ การปรับค่าดังกล่าวนี้ขึ้นอยู่กับความสามารถในการเรียนรู้ของโครงข่ายนั้นๆ และค่าที่แท้จริงของเอาต์พุตจะขึ้นอยู่กับฟังก์ชันการถ่ายโอน เพราะฉะนั้นการเลือกฟังก์ชันการถ่ายโอนให้เหมาะสมจึงเป็นสิ่งที่สำคัญอย่างยิ่ง

ฟังก์ชันการถ่ายโอน (Transfer Function) อาจเป็นฟังก์ชันแบบเชิงเส้น (Linear Function) หรือเป็นฟังก์ชันแบบไม่เป็นเชิงเส้น (Non-Linear Function) ก็ได้ ฟังก์ชันการถ่ายโอนมีให้เลือกใช้อย่างมากมาย แต่มีฟังก์ชันการถ่ายโอนอยู่ 3 รูปแบบที่ถูกเลือกใช้อยู่เสมอได้แก่ ฟังก์ชันแบบ Hard Limit (hardlim(n)), Linear (purelin(n)) และ Log-Sigmoid (logsig(n)) เป็นต้น

3.2.2 Multi-Input Neuron เป็นแบบจำลองนิวรอนที่มีจำนวนอินพุตมากกว่า 1 อินพุต ซึ่งเป็นเวกเตอร์อินพุต สามารถเขียนได้ดังนี้ $p_1, p_2, \dots, p_R$ โดยที่ R แทนจำนวนของสมาชิกในเวกเตอร์อินพุต สมาชิกทุกตัวในเวกเตอร์อินพุตคูณกับค่าถ่วงน้ำหนัก $w_{1,1}, w_{1,2}, \dots, w_{1,R}$ จากนั้นค่าทั้งหมดจะถูกนำมารวมกันที่ Summing Junction ถ้ามีค่าไบแอสก็จะถูกนำมารวมด้วย ณ จุดนี้ค่าผลรวมที่ได้คือค่า Net Input หลังจากนั้นจะส่งไปยังฟังก์ชันการถ่ายโอน เพื่อหาค่าเอาต์พุต สามารถเขียนให้อยู่ในรูปสมการได้ดังนี้

$$n = w_{1,1}p_1 + w_{1,2}p_2 + \dots + w_{1,R}p_R + b \quad (5)$$

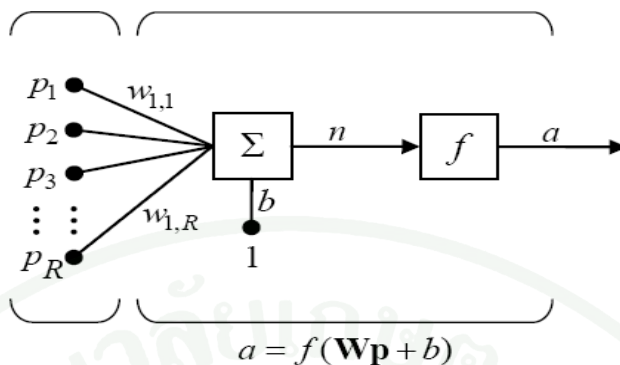
หรือ

$$n = Wp + b \quad (6)$$

หาค่าเอาต์พุต

$$a = f(Wp + b) \quad (7)$$

Inputs Multiple-Input Neuron

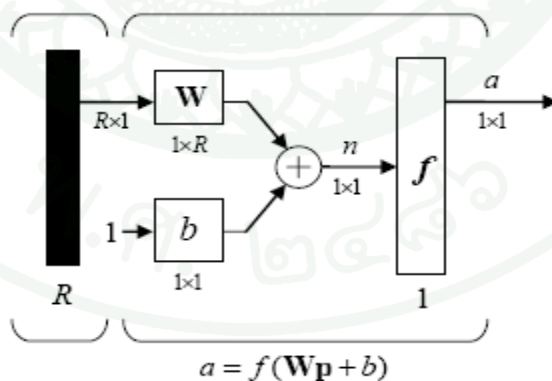


ภาพที่ 14 นิวรอนที่มีหลายอินพุต

ที่มา: Howard *et al.* (1996)

จากภาพที่ 14 ของนิวรอนเพียงนิวรอนเดียวข้างบน ประกอบไปด้วยรายละเอียดที่มากมายเมื่อพิจารณาถึงโครงข่ายที่ประกอบไปด้วยนิวรอนจำนวนมาก มีจำนวนของชั้นหลายๆ ก็ยังมีรายละเอียดที่มากขึ้นไปอีก ดังนั้นจึงมีการใช้สัญลักษณ์ย่อ (Abbreviated notation) เพื่อทำให้ง่ายต่อความเข้าใจแสดงดังภาพที่ 15

Input Multiple-Input Neuron

ภาพที่ 15 นิวรอนที่มีอินพุต R อินพุตแบบย่อที่มา: Howard *et al.* (1996)

จากภาพที่ 15 ค่าอินพุตเวกเตอร์ แทนด้วยแท่งสี่เหลี่ยมทึบแนวตั้งทางด้านซ้ายมือ มีขนาดของเวกเตอร์เท่ากับ $R \times 1$ นำไปคูณกับเวกเตอร์ของค่าถ่วงน้ำหนัก ซึ่งมีขนาดเท่ากับ $1 \times R$ แล้วนำค่าไบแอสที่มีขนาดเท่ากับ 1×1 มาบวกรวมไปด้วยที่ Summing Junction ก่อนที่จะส่งผลรวมนี้ ซึ่งมีขนาดเท่ากับ 1×1 ไปยังฟังก์ชันการถ่ายโอนเพื่อให้ได้ค่าเอาต์พุตออกมา ในกรณีนี้ค่าเอาต์พุตที่ได้จะมีขนาดเท่ากับ 1×1 เป็นปริมาณสเกลาร์ แต่ถ้ามีจำนวนนิวรอนมากกว่า 1 นิวรอน ค่าเอาต์พุตที่ได้จะเป็นปริมาณเวกเตอร์

3.3 การทำงานของโครงข่ายประสาทเทียม แบ่งเป็น 2 ขั้นตอนคือ

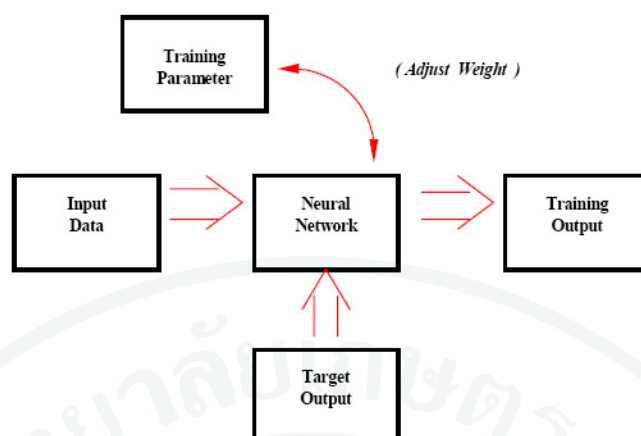
3.3.1 ขั้นตอนการเรียนรู้ ซึ่งเรียนรู้จากข้อมูลอินพุต และข้อมูลเอาต์พุต แล้วทำการออกแบบโครงข่ายประสาทเทียม เพื่อหารูปแบบและความสัมพันธ์ภายในของข้อมูลอินพุต และเอาต์พุต

3.3.2 ขั้นตอนการทดสอบ จะนำโครงข่ายประสาทเทียมที่ได้ออกแบบและฝึกสอนแล้ว มาใช้งานต่อโดยจะใช้อินพุตชุดใหม่ใส่เข้าไป (มีไม่มากกว่า 70% ของชุดข้อมูลที่ใช้ในการฝึกสอน) เพื่อหาเอาต์พุตที่ต้องการ

3.4 การเรียนรู้สำหรับโครงข่ายประสาทเทียม

3.4.1 การเรียนแบบมีการสอน (Supervised Learning)

เป็นการเรียนแบบที่มีการตรวจคำตอบเพื่อให้วงจรช่วยปรับตัว ชุดข้อมูลที่ใช้สอน วงจรข่ายจะมีคำตอบไว้คอยตรวจดูว่าวงจรข่ายให้คำตอบที่ถูกหรือไม่ ถ้าตอบไม่ถูก วงจรข่ายก็จะปรับตัวเองเพื่อให้ได้คำตอบที่ดีขึ้น (เปรียบเทียบกับคน เหมือนกับการสอนนักเรียนโดยมีครูผู้สอนคอยแนะนำ)

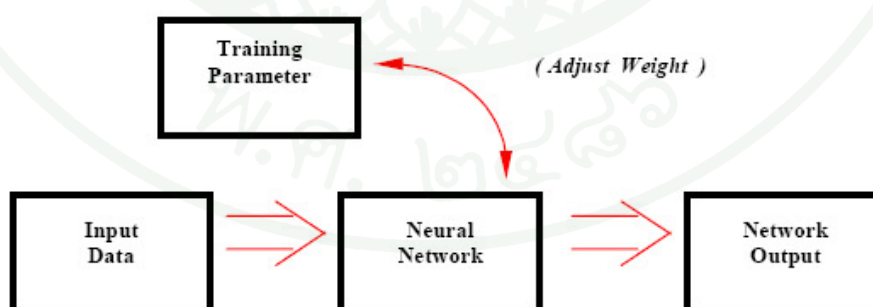


ภาพที่ 16 ลักษณะการเรียนรู้แบบมีการสอน (Supervised Learning)

ที่มา: ชูพิץ (2551)

3.4.2 Unsupervised Learning การเรียนรู้แบบไม่มีการสอน

เป็นการเรียนรู้แบบไม่มีผู้แนะนำ ไม่มีการตรวจคำตอบว่าถูกหรือผิด วงจรข่ายจะจัดเรียงโครงสร้างด้วยตัวเองตามลักษณะของข้อมูล ผลลัพธ์ที่ได้ วงจรข่ายจะสามารถจัดหมวดหมู่ของข้อมูลได้ (เปรียบเทียบกับคน เช่น การที่เราสามารถแยกแยะพันธุ์พืช พันธุ์สัตว์ตามลักษณะรูปร่างของมันได้เองโดยไม่มีใครสอน)

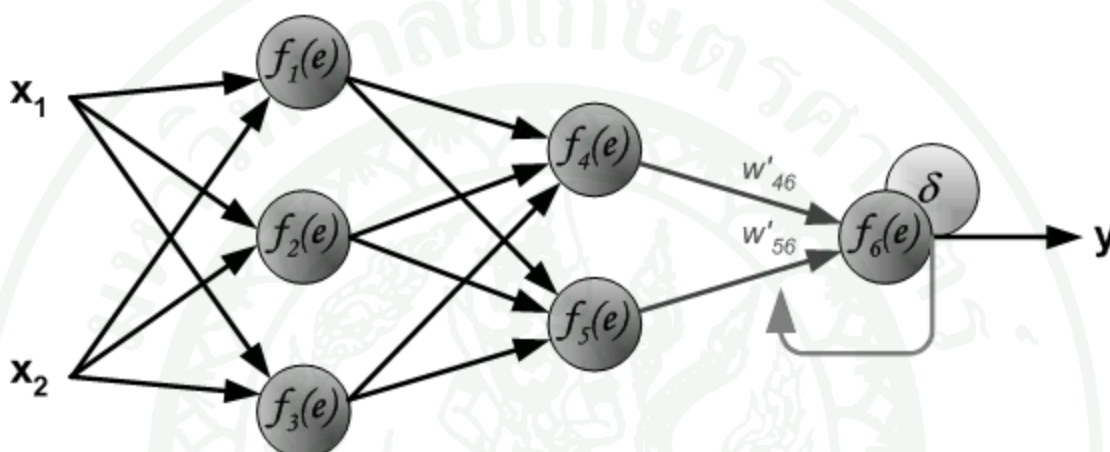


ภาพที่ 17 แสดงการเรียนรู้แบบไม่มีการสอน (Unsupervised Learning)

ที่มา: ชูพิץ (2551)

3.5 แบบจำลองโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ (Back-propagation Neural Network)

การแพร่ค่าย้อนกลับเป็นอัลกอริทึมที่ใช้ในการเรียนรู้ของเครือข่ายประสาทวิธีหนึ่ง ที่นิยมใช้ใน multilayer perceptron เพื่อปรับค่าน้ำหนักในเส้นเชื่อมต่อระหว่างโหนดให้เหมาะสม โดยการปรับค่านี้อาจขึ้นกับความแตกต่างของค่าเอาต์พุตที่คำนวณได้กับค่าเอาต์พุตที่ต้องการ พิจารณารูปต่อไปนี้ประกอบ



ภาพที่ 18 แสดงรูปแบบ Back- propagation neural network

ลำดับขั้นการประมวลผลของโครงข่ายประสาทเทียม

ขั้นที่ 1 ชุดข้อมูลที่ใช้ในการฝึกโครงข่ายประสาทเทียม (Training Set)

ในการเรียนรู้ของโครงข่ายประสาทเทียม โดยอาศัยการแนะนำ จำเป็นต้องใช้ชุดข้อมูลอินพุต และข้อมูลเป้าหมายในการฝึกโครงข่ายประสาทเทียม เพื่อให้โครงข่ายดังกล่าวสร้างสัญญาณออกได้สอดคล้องกับสัญญาณเป้าหมายเมื่อโครงข่ายประสาทเทียม ได้รับข้อมูลสัญญาณเข้าลักษณะเดียวกัน ชุดข้อมูลที่ใช้ในการฝึกโครงข่ายสามารถเขียนแทนได้ดังสมการที่ 8

$$P = \{(X(1), T(1)), (X(2), T(2)), \dots, (X(N), T(N))\} \quad (8)$$

โดยที่ P คือชุดของข้อมูลที่ใช้ในการฝึกโครงข่าย (Training Set)

N คือจำนวนชุดข้อมูลที่ใช้ในการฝึกโครงข่าย (Training Pair)

ขั้นที่ 2 การกำหนดค่าเริ่มต้นสำหรับพารามิเตอร์ของโครงข่ายประสาทเทียม

การฝึกโครงข่ายประสาทเทียม จำเป็นต้องกำหนดค่าเริ่มต้นให้กับค่าน้ำหนัก (Weight) และค่าไบแอส (Bias) รวมทั้งค่าอัตราการเรียนรู้ (Learning rate) เพื่อใช้เป็นจุดเริ่มต้นในการประมวลผลของโครงข่ายซึ่งจะกล่าวต่อไป

ขั้นที่ 3 กระบวนการประมวลผลของโครงข่าย

กระบวนการประมวลผลของโครงข่ายประสาทเทียม แบ่งเป็น 2 ตอน

ตอนที่ 1 การคำนวณไปข้างหน้า (Feedforward Calculation)

การคำนวณเริ่มจากชั้นอินพุต ผ่านชั้นซ่อนภายในจนถึงชั้น เอาต์พุตแสดงดังสมการที่

9 - 12

พิจารณาชั้นอินพุต (Input Layer)

ปมนิวรอนในชั้นอินพุตจะรับข้อมูลอินพุต (X) จากชุดข้อมูลที่ใช้ในการฝึกโครงข่าย (Training Set) และส่งต่อไปยังชั้นภายในเริ่มทำการคำนวณต่อไป

พิจารณาชั้นซ่อนภายใน (Hidden Layer)

สัญญาณที่เข้าสู่ชั้นซ่อนภายในสามารถแสดงได้ดังสมการที่ 9

$$Z_{in} = V^T X + V_{Bias} \quad (9)$$

สัญญาณที่ออกมาจากชั้นซ่อนภายในสามารถแสดงได้ดังสมการที่ 10

$$Z = f(Z_{in}) \quad (10)$$

พิจารณาสัญญาณเอาต์พุต (Output Layer)

สัญญาณที่เข้าสู่เอาต์พุตสามารถแสดงได้ดังสมการที่ 11

$$Y_{in} = W^T Z + W_{Bias} \quad (11)$$

สัญญาณที่ออกจากชั้นเอาต์พุตสามารถแสดงได้ดังสมการที่ 12

$$Y = f(Y_{in}) \quad (12)$$

ตอนที่ 2 การแพร่ค่าความคลาดเคลื่อนย้อนกลับ (Backpropagation of Error)

การปรับค่าน้ำหนักของโครงข่ายประสาทเทียม เพื่อลดค่าความคลาดเคลื่อนหรือความแตกต่างระหว่างข้อมูลเป้าหมาย $T(k)$ กับสัญญาณออกจากโครงข่ายประสาทเทียม ในชั้นเอาต์พุต $Y(k)$ ใช้วิธีการออปติไมซ์แบบเกรเดียนต์ โดยกำหนดฟังก์ชันความคลาดเคลื่อน (Error Function : $E(k)$) ดังสมการที่ 13 ในที่นี้พิจารณาในกรณีปมนิวรอนในชั้นเอาต์พุตมีจำนวน 1 ปม เนื่องจากระบบควบคุมที่นำไปใช้เป็นระบบอินพุตเอาต์พุตเดี่ยว (Single Input and Single Output : SISO System)

$$E(k) = \frac{1}{2} [T(k) - Y(k)]^2 \quad (13)$$

เมื่อ $E(k)$ แทนฟังก์ชันความคลาดเคลื่อนของชุดข้อมูลที่ k

คำนวณการลดค่าความคลาดเคลื่อนเทียบกับค่าน้ำหนัก และค่าไบแอสของโครงข่ายประสาทเทียม ที่เชื่อมอยู่ระหว่างชั้นซ่อนภายในกับชั้นเอาต์พุต แสดงในสมการที่ 14

$$\begin{aligned} \frac{\partial E}{\partial W} &= \frac{\partial [0.5 \{T(k) - Y(k)\}^2]}{\partial W} \\ &= \frac{\partial [0.5 \{T(k) - f(Y_{in})\}^2]}{\partial W} \\ &= -[T(k) - Y(k)] \frac{\partial f(Y_{in})}{\partial W} \\ \frac{\partial E}{\partial W} &= -[T(k) - Y(k)] f'(y_{in}) Z \end{aligned} \quad (14)$$

พิจารณาชั้นเอาต์พุต (Output Layer)

จากสมการที่ 10 กำหนด δ_o เป็นความสัมพันธ์แสดงดังสมการที่ 15

$$\delta_o = [T(k) - Y(k)] f'(y_{in}) \quad (15)$$

การเปลี่ยนแปลงค่าน้ำหนักของแขนซึ่งเชื่อมระหว่างชั้นซ่อนภายในกับชั้นเอาต์พุต (ΔW) แสดงดังสมการที่ 16-17

$$\Delta W = -\alpha \frac{\partial E}{\partial W} \quad (16)$$

หรือ

$$\Delta W = \alpha Z \delta_o^T \quad (17)$$

ในทำนองเดียวกัน การเปลี่ยนแปลงค่าไบแอสของปมนิวรอนในชั้นเอาต์พุต (ΔW_{Bias}) แสดงดังสมการที่ 18

$$\Delta W_{Bias} = \alpha \delta_o \quad (18)$$

พิจารณาชั้นซ่อนภายใน (Hidden Layer)

การลดค่าความคลาดเคลื่อนเทียบกับค่าน้ำหนักและค่าไบแอสที่เชื่อมระหว่างชั้นอินพุตกับชั้นซ่อนภายในพิจารณาทำนองเดียวกับชั้นเอาต์พุต ดังแสดงในสมการที่ 19

$$\begin{aligned} \frac{\partial E}{\partial V} &= \frac{\partial [0.5\{T(k) - Y(k)\}^2]}{\partial V} \\ &= -[T(k) - Y(k)] f'(y_{in}) \frac{\partial(Y_{in})}{\partial V} \\ &= -\delta_o \frac{\partial(Y_{in})}{\partial V} \end{aligned}$$

$$\frac{\partial E}{\partial V} = -\delta_o W f'(Z_{in}) X \quad (19)$$

เมื่อกำหนดให้ δ_h เป็นความล้มพันธ์แสดงดังสมการที่ 20

$$\delta_h = (W \delta_o) f'(Z_{in}) \quad (20)$$

การเปลี่ยนแปลงค่าน้ำหนักที่เชื่อมระหว่างชั้นอินพุตกับชั้นซ่อนภายใน (ΔV) แสดงดังสมการที่ 21 - 22

$$\Delta V = -\alpha \frac{\partial E}{\partial V} \quad (21)$$

หรือ

$$\Delta V = \alpha X \delta_h^T \quad (22)$$

ในทำนองเดียวกัน การเปลี่ยนแปลงค่าไบแอสของปมนิวรอนในชั้นซ่อนภายใน (ΔV_{Bias}) แสดงดังสมการที่ 23

$$\Delta V_{Bias} = \alpha \delta_h \quad (23)$$

ขั้นที่ 4 การปรับค่าน้ำหนัก (Weight) และค่าไบแอส (Bias)

หลังจากการประมวลผลในขั้นที่ 3 โครงข่ายประสาทเทียม จะปรับฐานข้อมูลความรู้ภายในใหม่ ซึ่งอยู่ในรูปของการปรับค่าน้ำหนัก และค่าน้ำหนักไบแอสในแต่ละชั้นให้ทันสมัยและสอดคล้องกับชุดข้อมูลความรู้ใหม่ที่ได้เรียนรู้ เป็นผลให้ฟังก์ชันความคลาดเคลื่อนดังสมการที่ 20 ลดลง

พิจารณาการปรับค่าน้ำหนัก และค่าไบแอสของแขนที่เชื่อมระหว่างชั้นซ่อนภายในกับชั้นเอาต์พุต แสดงดังสมการที่ 24-24

$$\Delta V(k+1) = V(k) + \Delta V(k) \quad (24)$$

$$\Delta V_{Bias}(k+1) = V_{Bias}(k) + \Delta V_{Bias}(k) \quad (25)$$

พิจารณาการปรับค่าน้ำหนักและค่าไบแอสของแกนที่เชื่อมระหว่างชั้นซ่อนภายในกับชั้นเอาต์พุต ดังสมการที่ 26-27

$$\Delta W(k+1) = W(k) + \Delta W(k) \quad (26)$$

$$\Delta W_{Bias}(k+1) = W_{Bias}(k) + \Delta W_{Bias}(k) \quad (27)$$

ขั้นที่ 5 ตรวจสอบจำนวนชุดข้อมูลที่ใช้ฝึกโครงข่าย

ข้อมูลที่ใช้ในการฝึกโครงข่าย (Training Set :P) จะถูกป้อนเข้ามาครั้งละ 1 คู่ลำดับ (Training Pair: {X(k),T(k)}) ซึ่งประกอบไปด้วย ข้อมูลอินพุต (Input Data: X(k)) และข้อมูลเป้าหมาย (Target Data :T(k)) แต่เนื่องจากชุดข้อมูลที่ใช้ในการฝึกโครงข่ายมีจำนวนทั้งสิ้น N ชุด ดังนั้นจำเป็นต้องป้อนข้อมูลที่ใช้ในการฝึกโครงข่ายได้รับมาครบหรือไม่ ถ้าไม่ครบตามจำนวน จำเป็นต้องป้อนข้อมูลชุดถัดมา แล้วจึงเริ่มทำการประมวลผลตามขั้นตอนที่ 3 อีกครั้ง แต่ถ้าจำนวนข้อมูลที่ใช้ฝึกครบตามจำนวนจะผ่านไปยังขั้นตอนที่ 6 ต่อไป

ขั้นที่ 6 ตรวจสอบดัชนีสมรรถนะ (Performance Index)

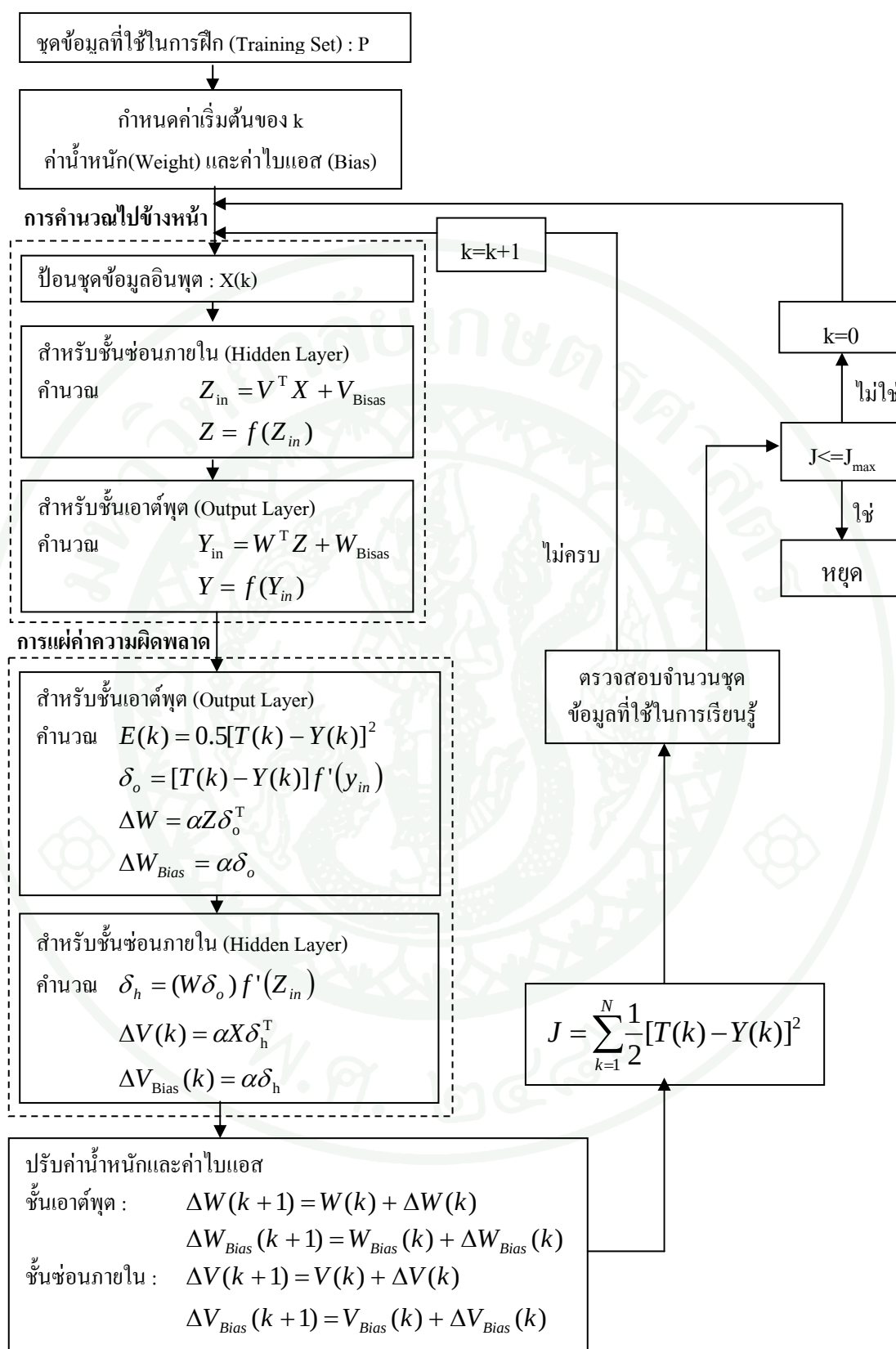
วัตถุประสงค์หลักของการฝึกโครงข่ายประสาทเทียมคือ การลดฟังก์ชันความคลาดเคลื่อนระหว่างข้อมูลที่ออกจากโครงข่ายประสาทเทียมและข้อมูลเป้าหมาย ดรรชนีสมรรถนะของการฝึกโครงข่ายประสาทเทียมแสดงดังสมการที่ 28

$$J = \sum_{k=1}^N \frac{1}{2} [T(k) - Y(k)]^2 \quad (28)$$

โดยที่ J ดรรชนีสมรรถนะของการฝึกโครงข่ายประสาทเทียม ถ้าดรรชนีสมรรถนะ (J) มีค่าต่ำกว่าที่กำหนด (J_{max}) แสดงว่าโครงข่ายประสาทเทียมได้รับการฝึกให้สามารถเรียนรู้และ

สามารถสร้างผลตอบได้สอดคล้องกับข้อมูลที่ป้อนให้เรียนรู้ ในทางตรงกันข้ามถ้ากรณีที่
สมรรถนะ (J) มีค่ามากกว่าที่กำหนด ($J_{\max}$) จะเริ่มทำการฝึกตามขั้นที่ 3 อีกครั้ง จากการประมวลผล
ทั้ง 6 ขั้นตอน สามารถสรุปได้ดังภาพที่ 19





ภาพที่ 19 แสดงลำดับการประมวลผลของโครงข่าย

3.6 การประยุกต์ใช้โครงข่ายประสาทเทียม

ในส่วนนี้จะเป็นการกล่าวถึงการนำโครงข่ายประสาทเทียมไปประยุกต์ใช้ในการแก้ปัญหาสาขาต่างๆ ได้อย่างหลากหลาย และมีประสิทธิภาพเป็นที่ยอมรับกันอย่างกว้างขวาง มีดังต่อไปนี้

3.6.1 Constraint Satisfaction คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้เพื่อหาคำตอบของปัญหาต่างๆ แต่ต้องอยู่ภายใต้ข้อจำกัด หรือเงื่อนไขที่กำหนด เช่น การจัดตารางการทำงานของเครื่องจักรที่ต้องทำงานเพียงอย่างเดียว การเดินทางหากรุกที่มีรูปแบบการเดินทางที่แตกต่างกัน และการเจาะรูบนแผ่นปริ้นส์อย่างไรโดยใช้เวลาน้อยที่สุด เพื่อลดต้นทุนในการผลิต เป็นต้น

3.6.2 Content Address Memories คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้เพื่อการจัดเก็บข้อมูลของหน่วยความจำบนเครื่องคอมพิวเตอร์ โดยเลือกใช้สถาปัตยกรรมของโครงข่ายประสาทเทียม ที่สามารถเรียนรู้รูปแบบการจำ และการเก็บข้อมูล ถ้ารูปแบบการจำ และการเก็บข้อมูลเหมือนกันจะเรียกว่า “Autoassociative retrieval” แต่ถ้ารูปแบบการจำ และการเก็บข้อมูลไม่เหมือนกันจะเรียกว่า “Heteroassociative retrieval”

3.6.3 Control คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้ในงานควบคุมที่มีความซับซ้อน เช่น การควบคุมการทำงานของหุ่นยนต์ การควบคุมการขับเคลื่อนสิ่งต่างๆ โดยไม่ต้องใช้มนุษย์ควบคุม เช่นรถยนต์ เครื่องบิน เป็นต้น การควบคุมตำแหน่งของแท่งอิเล็กทรอนิกส์ขนาดใหญ่ในการเชื่อมไฟฟ้าเพื่อลดการใช้พลังงานไฟฟ้า และยืดอายุการใช้งานของอุปกรณ์นั้นๆ การควบคุมปริมาณของสารเคมีที่ใช้ในกระบวนการต่างๆ เป็นต้น

3.6.4 Data Compression คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้ในการลดขนาดข้อมูล ในบางระบบ ต้องใช้การเรียนรู้ของโครงข่ายประสาทเทียม เพื่อการลดขนาดข้อมูลในระบบการส่งข้อมูล และการจัดเก็บข้อมูล ทำให้ข้อมูลที่มีลักษณะที่เหมือนกันสามารถถูกส่งผ่านไปพร้อมกันได้ หรือจัดเก็บไว้ด้วยกันได้ เป็นต้น

3.6.5 Diagnostics คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้ในการวิเคราะห์หรือวินิจฉัยสิ่งที่ผิดปกติในสาขาแพทย์ วิศวกรรม และอุตสาหกรรมการผลิตต่างๆ เช่น การ

วินิจฉัย โรคของแพทย์ การวิเคราะห์การทำงานที่ผิดพลาดของงานประเภท และอุปกรณ์ต่างๆ เป็นต้น

3.6.6 Forecasting คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้ในการพยากรณ์ที่จะเกิดขึ้นในอนาคต โดยใช้ข้อมูลในอดีตที่เหมาะสม เช่น การพยากรณ์ปริมาณการเติบโตของการขายสินค้าใหม่ๆ เพื่อใช้วางแผนการผลิต การพยากรณ์อากาศสำหรับนักอุตุนิยมวิทยา การพยากรณ์ความน่าเชื่อถือทางการเงินของธนาคาร ในการพิจารณาการกู้ยืมเงินของบริษัทต่างๆ การพยากรณ์ปริมาณการเติบโตของผู้ใช้สนามบิน เพื่อการจัดการบริหารสนามบินที่ดี การพยากรณ์ความต้องการไฟฟ้าเพื่อให้สามารถวางแผนการผลิตไฟฟ้าให้เพียงพอในอนาคต การพยากรณ์อัตราแลกเปลี่ยนเงินตราต่างประเทศ การพยากรณ์การซื้อขายหุ้น เป็นต้น

3.6.7 General Mapping คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้เพื่อแปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสม เช่น การนำข้อมูลออกจากหน่วยความจำ การแปลงข้อมูลที่เกิดจากการบีบอัดข้อมูล การแปลงสัญญาณ เป็นต้น

3.6.8 Multisensor Data Fusion คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้ในการรวมสัญญาณที่มาจากแหล่งกำเนิดเพื่อให้ได้รับข้อมูลได้มากกว่าการรับสัญญาณจากแหล่งกำเนิดเดียว กระบวนการ Fusion ประกอบด้วย การตรวจสอบ (Detection) การเชื่อมโยง (Association) การหาความสัมพันธ์ (Correlation) การประมาณ (Estimation) และการรวมกัน (Combination) ของข้อมูล ใช้ในการประมวลผล การแสดงผล การสร้างหุ่นยนต์ การประเมินสถานการณ์ทางทหาร การวินิจฉัยโรคของแพทย์ เป็นต้น

3.6.9 Optimization คือการนำโครงข่ายประสาทเทียมมาประยุกต์ใช้ในการหาคำตอบที่เหมาะสม ภายใต้เงื่อนไขหรือข้อจำกัดที่กำหนด เช่น การกำหนดราคาขาย และการลดราคาตัวเครื่องบินของสายการบิน การจัดการตารางการทำงานของกระบวนการผลิตให้มีประสิทธิภาพ การหาระยะทางที่สั้นที่สุดในการเดินทาง เป็นต้น

3.6.10 Pattern Recognition คือการนำโครงข่ายประสาทเทียมมาประยุกต์ในการเรียนรู้ และจำรูปแบบต่างๆ ที่มีความซับซ้อนได้ เช่น การเรียนรู้และจำภาพของวัตถุต่างๆ การเรียนรู้และจำตัวอักษรแบบพิมพ์ หรือตัวอักษรที่เป็นลายมือ การเรียนรู้และจำเสียงพูด เป็นต้น

3.6.11 Risk Assessment คือการนำโครงข่ายประสาทเทียมมาประยุกต์ในการประเมินความเสี่ยง การประยุกต์นี้ต้องเรียนรู้และจำรูปแบบของเหตุการณ์ที่มีความเสี่ยงสูง และมีความเสี่ยงต่ำ แล้วใช้ในการแยกความแตกต่าง จัดกลุ่มของเหตุการณ์ เช่น การประเมินความเสี่ยงทางเศรษฐศาสตร์ การประเมินความเสี่ยงในการที่อุปกรณ์จะเกิดความเสียหาย เป็นต้น

4. การควบคุมโดยโครงข่ายประสาท (Neural Control)

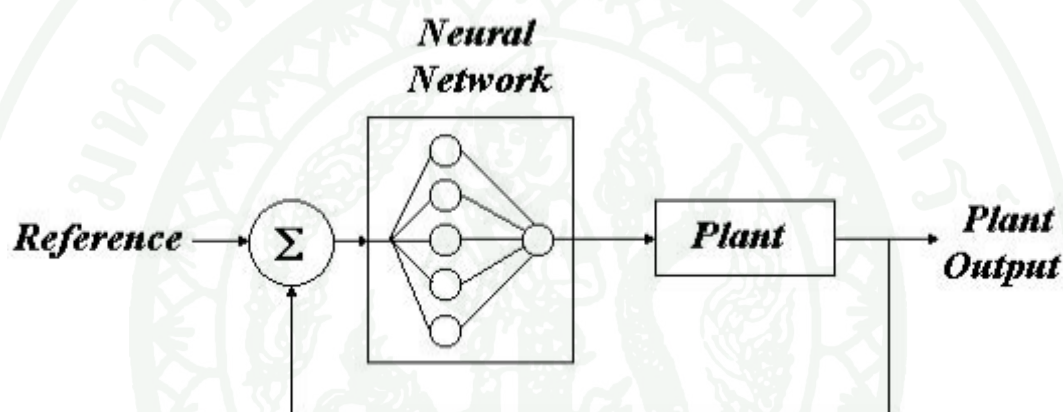
Neural Control หมายถึงวิธีการควบคุมโดยใช้โครงข่ายประสาท และวิธีการออกแบบระบบอยู่ในพื้นฐานโครงข่ายประสาทของแพลนท์ โดยทั่วไปสำหรับการใช้โครงข่ายประสาทในการควบคุมจะมีทั้งวิธีทางตรงและทางอ้อมในการออกแบบระบบควบคุม

โดยทั่วไปของโครงข่ายประสาท คือการประมาณหน้าที่ของอุปกรณ์ โดยเฉพาะเมื่อเราไม่ทราบแบบจำลองทางคณิตศาสตร์ของแพลนท์ โครงข่ายประสาทจะทำให้สามารถออกแบบตัวควบคุมได้ โดยเราต้องมีข้อมูลเกี่ยวกับระบบในรูปแบบของข้อมูลอินพุต เอาต์พุต ในอีกแง่หนึ่งคือโครงข่ายประสาทสามารถเปรียบเสมือนกล่องดำ (Black box) ของแบบจำลองระบบ ดังนั้นตัวควบคุมโครงข่ายประสาทจะได้จากการเรียนรู้โครงข่ายประสาท และความสามารถที่เหมาะสมสำหรับการปรับตัวควบคุมต้องปรับให้เข้ากับการเปลี่ยนแปลงสภาพแวดล้อม เช่นระบบที่เปลี่ยนแปลงไปตามเวลา ในทางปฏิบัติตัวควบคุมโครงข่ายประสาทได้พิสูจน์ว่ามีประโยชน์อย่างยิ่งสำหรับระบบที่ไม่เปลี่ยนแปลงไปตามเวลาด้วย

เพื่อสร้างตัวควบคุมโครงข่ายประสาทที่สามารถบังคับระบบให้เป็นไปตามที่ต้องการเราต้องปรับพารามิเตอร์จากข้อผิดพลาดที่สังเกตได้ ซึ่งเกิดจากความแตกต่างระหว่างเอาต์พุตของระบบกับเอาต์พุตที่ต้องการ การปรับพารามิเตอร์ของตัวควบคุมจะทำได้โดยวิธี Propagating Back ความผิดพลาดจะเชื่อมโยงกันภายในโครงสร้างของโครงข่ายประสาทเอง ซึ่งเป็นไปได้ถ้าทราบแบบจำลองทางคณิตศาสตร์ของระบบ แต่ถ้าไม่ทราบแบบจำลองทางคณิตศาสตร์ของระบบอย่างน้อยต้องทราบแบบจำลองโดยประมาณของระบบเพื่อใช้วิธี Propagating Back ข้างต้น การประมาณแบบจำลองของระบบ เรียกว่า Identified Model เมื่อเราป้อนข้อมูลอินพุต เอาต์พุตของระบบเพื่อฝึกโครงข่ายประสาทให้แบบจำลองประมาณของระบบ เราจะได้รับ Neural Network Identified Model ของระบบ ซึ่งวิธีนี้มีการใช้ในการควบคุมโดยโครงข่ายประสาททางอ้อม

4.1 Neural networks in direct neural control

การประมาณหน้าที่ของการควบคุมสามารถทำได้หลายวิธี เมื่อเราใช้โครงข่ายประสาทสำหรับแบบจำลองพลศาสตร์ย้อนกลับ (Inverse Dynamics) นั้นหมายถึงเรากำลังใช้การออกแบบตัวควบคุมโครงข่ายประสาทโดยตรง การออกแบบทางตรง หมายความว่าโครงข่ายประสาทดำเนินการโดยตรงกับตัวควบคุม โครงข่ายประสาทจะต้องผ่านการฝึกเป็นตัวควบคุมตามบางคุณสมบัติ โดยใช้ข้อมูลอินพุต เอาต์พุตที่เป็นข้อมูลตัวเลข หรือใช้แบบจำลองทางคณิตศาสตร์ของระบบก็ได้ การออกแบบตัวควบคุมโครงข่ายประสาทโดยตรงนั้นแสดงในภาพที่ 20



ภาพที่ 20 ตัวควบคุมโครงข่ายประสาทโดยตรง

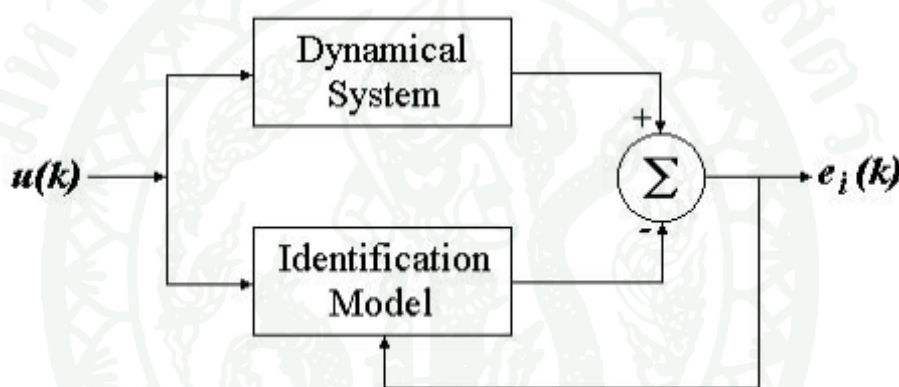
4.2 Neural networks in indirect neural control

การออกแบบตัวควบคุมโครงข่ายประสาททางอ้อมเป็นไปตาม แบบโครงข่ายประสาทของระบบที่จะควบคุม ในกรณีนี้ตัวควบคุมเองจะไม่อาจเป็นโครงข่ายประสาท แต่จะมาจากระบบซึ่งถูกจำลองโดยโครงข่ายประสาท ลักษณะนี้คล้ายกับการควบคุมมาตรฐานในแบบจำลองทางคณิตศาสตร์ ซึ่งแบบจำลองทางคณิตศาสตร์จะกลายเป็นโครงข่ายประสาท

การออกแบบตัวควบคุมโครงข่ายประสาททางอ้อมจะเกี่ยวข้องกับสองส่วน ส่วนแรกประกอบด้วย การระบุการเปลี่ยนแปลงระบบโดยโครงข่ายประสาทจากข้อมูลการฝึก ที่เรียกว่า System Identification ในส่วนที่สอง การออกแบบการควบคุมค่อนข้างจะง่าย แม้ตัวควบคุมที่ได้มานั้นไม่ใช่มาจากมาตรฐานแบบจำลองทางคณิตศาสตร์ของระบบ แต่จะมาจาก Neural Network

Identified Model ซึ่ง Neural Network Identified Model ของระบบนั้นจะไม่ได้เป็นแบบเชิงเส้น วิธีหนึ่งสำหรับการออกแบบตัวควบคุมแบบนี้เป็นคือ การทำให้ Neural Network Identified Model เป็นเชิงเส้น (Linearize) และใช้การออกแบบตัวควบคุมเชิงเส้นแบบพื้นฐานทั่วไป

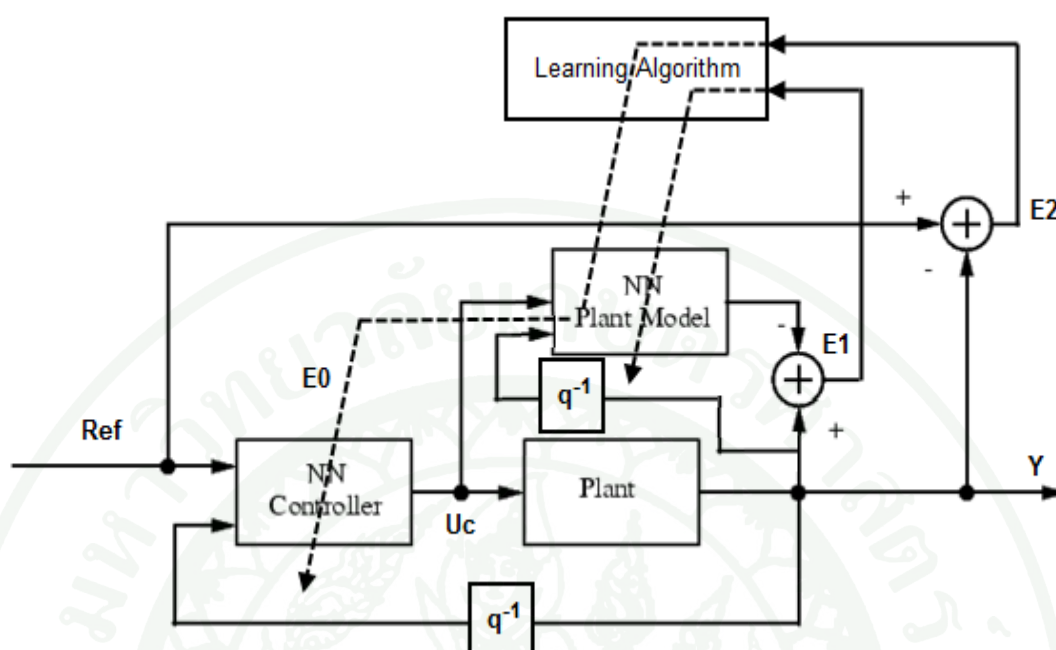
จากภาพที่ 21 ปัญหาทั่วไปของ System Identification ในระบบไม่เชิงเส้น พารามิเตอร์ของ Identified Model จะถูกประมาณเป็นแบบจำลองที่มีการเปลี่ยนแปลงตลอดเวลา เพื่อให้เกิดความแตกต่างระหว่างเอาต์พุตของระบบกับแบบจำลองเอาต์พุตตลอดเวลาน้อยที่สุด โครงข่ายประสาทจะเหมาะสมสำหรับ System Identification ที่ไม่ทราบแบบจำลองทางคณิตศาสตร์และทราบแค่การทำงานของระบบในรูปแบบของข้อมูลตัวอย่างอินพุตเอาต์พุต



ภาพที่ 21 System Identification Model

4.3 การควบคุมแบบปรับตัวเองโดยใช้โครงข่ายประสาทเทียม

เนื่องจากความสามารถของโครงข่ายประสาทในการเรียนรู้ จดจำลักษณะความสัมพันธ์ของข้อมูล และสามารถให้ผลตอบที่สอดคล้องกับชุดของข้อมูลที่ป้อนเข้าหลังจากที่ได้รับการฝึกฝนอย่างพอเพียง จึงสามารถนำไปประยุกต์ใช้กับการควบคุมแบบปรับตัวเองได้ เนื่องจากไม่ทราบสัญญาณควบคุมที่เหมาะสมในการป้อนเข้าสู่ระบบเพื่อให้ระบบเป็นไปตามต้องการ ดังนั้นการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมจึงอาศัยวิธีส่งผ่านค่าความคลาดเคลื่อนระหว่างสัญญาณอ้างอิงและสัญญาณที่ออกจากระบบ ผ่านกลับไปตามชั้นต่างๆของโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ (Neural Network Identification) ค่าความคลาดเคลื่อนที่แผ่กลับมาถูกใช้ในการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุม (Neural Network Controller) ให้สามารถเรียนรู้และลดค่าความคลาดเคลื่อนดังกล่าว ดังแสดงในรูปที่ 22



ภาพที่ 22 แผนแสดงการควบคุมแบบปรับตัวเองโดยใช้โครงข่ายประสาทเทียม

การประยุกต์โครงข่ายประสาทเทียมในการควบคุมแบบปรับตัวเองดังรูปที่ 22 ประกอบด้วยโครงข่ายประสาท 2 ส่วน คือ ส่วนของการหาเอกลักษณ์ และส่วนเป็นตัวควบคุม โดยส่วนการหาเอกลักษณ์จะทำหน้าที่ในการจำลองพฤติกรรมของระบบ โดยใช้ค่าความแตกต่างระหว่างสัญญาณเอาต์พุตออกจากพลานท์ (Plant Output Signal : Y) กับสัญญาณเอาต์พุตเครือข่ายจากโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ (Neural Network Identification Output Signal : Y_m) E_1 เป็นตัวสร้างสัญญาณที่ใช้ในการฝึกโครงข่ายประสาทที่ใช้เป็นตัวควบคุมโดยอาศัยการส่งผ่านค่าความคลาดเคลื่อนระหว่างสัญญาณอ้างอิง (Reference Signal : Ref) กับสัญญาณเอาต์พุตที่ออกจากพลานท์ E_2 แล้วย้อนกลับผ่านชั้นต่างๆจนถึงชั้นอินพุตของโครงข่ายประสาทการหาเอกลักษณ์ ค่าความเคลื่อนที่ปมนิวรอนในชั้นอินพุต (E_0) นำไปใช้ในการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุม เมื่อหลังจากได้โครงข่ายประสาทเทียมทั้งสองแล้วจะนำไปใช้กับระบบ โดยค่าพารามิเตอร์ของโครงข่ายประสาททั้งสองจะถูกปรับเพื่อลดค่าความคลาดเคลื่อนดังกล่าว และสร้างสัญญาณควบคุม (Control Signal : U_c) ป้อนให้กับพลานท์ เพื่อให้สมรรถนะของพลานท์เป็นไปตามสัญญาณอ้างอิงตามที่กำหนด โครงข่ายประสาทเทียมทั้งสองเป็นชนิดแพร่ค่าย้อนกลับ

การออกแบบโครงข่ายประสาทเทียมในส่วนการหาเอกลักษณ์

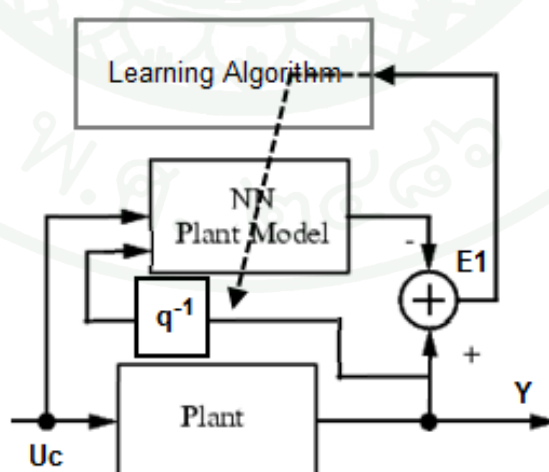
รูปแบบสมการการหาเอกลักษณ์โดยโครงข่ายประสาทเทียมแสดงดังสมการที่ 29

$$Y_m(k+1) = fNN_m(Y(k), U(k)) \quad (29)$$

โดยที่

$fNN_m\{\bullet\}$	โครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์
$(Y(k), U(k))$	ชุดข้อมูลอินพุตที่ป้อนเข้าสู่โครงข่ายประสาท
$Y_m(k+1)$	สัญญาณเอาต์พุตจากโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์
$Y(k)$	ชุดข้อมูลเอาต์พุตจากพลานท์
$U(k)$	ชุดข้อมูลอินพุตของพลานท์

รูปแบบจำลองดังกล่าว ไม่จำเป็นต้องทราบลักษณะสมการพลวัตที่แท้จริงของระบบ ดังนั้นวิธีการดังกล่าวจะไม่ทราบค่าพารามิเตอร์ที่แท้จริงของระบบแต่ค่าพารามิเตอร์ดังกล่าวจะแฝงอยู่ในรูปของค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาท ซึ่งนำไปใช้ในโครงข่ายประสาทที่เป็นตัวควบคุม จากสมการที่ 25 แสดงรูปการใช้โครงข่ายประสาทในการหาเอกลักษณ์ ได้ดังภาพที่ 23



ภาพที่ 23 แสดงการประยุกต์โครงข่ายประสาทในการหาเอกลักษณ์

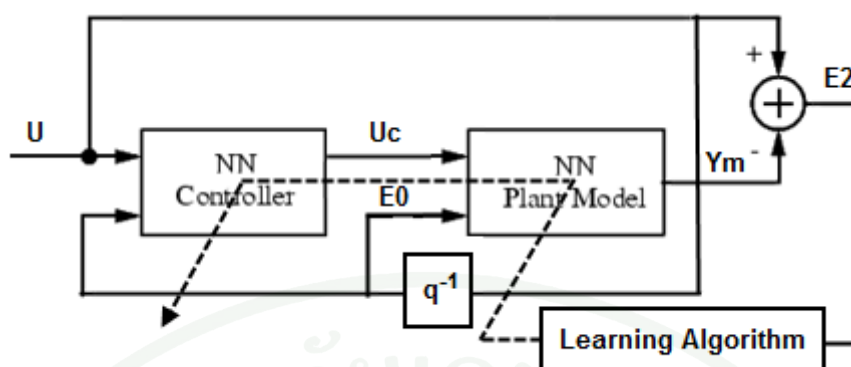
จากรูปที่ 23 การฝึกโครงข่ายประสาทในการหาเอกลักษณ์ประกอบไปด้วยชุดข้อมูลอินพุต ($Y(k), U(k)$) และชุดเป้าหมาย ($Y(k+1)$) ค่าความคลาดเคลื่อนระหว่างสัญญาณออกจากโครงข่ายประสาทและชุดข้อมูลเป้าหมาย E_1 ถูกใช้ในการปรับค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาท กำหนดครรชนีสมรรถนะของการหาเอกลักษณ์ด้วยโครงข่ายประสาท (JNN_m) ดังสมการที่ 30

$$JNN_m = \frac{1}{2} \sum_{k=0}^N (Y(k) - Y_m(k))^2 \quad (30)$$

โดยที่ N เป็นจำนวนชุดข้อมูลที่ใช้ในการฝึกโครงข่ายประสาท (Training Data)

การออกแบบโครงข่ายประสาทเพื่อเป็นตัวควบคุม

โดยทั่วไปการฝึกโครงข่ายประสาทจำเป็นต้องทราบค่าเป้าหมาย (Target Signal) โดยสัญญาณเป้าหมายในการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมคือ สัญญาณควบคุมที่แท้จริงที่ป้อนเข้าสู่ระบบเพื่อให้ผลตอบสนองเป็นไปตามสัญญาณอ้างอิง แต่ในทางปฏิบัติจะไม่ทราบค่าสัญญาณควบคุมที่เหมาะสม ดังนั้นการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมจึงใช้การฝึกทางอ้อม อาศัยการกำหนดครรชนีสมรรถนะของตัวควบคุมเป็นค่าความแตกต่างระหว่างสัญญาณอ้างอิงกับสัญญาณที่ออกจากโครงข่ายประสาทในการหาเอกลักษณ์ ค่าความคลาดเคลื่อนดังกล่าวจะถูกส่งไปยังปมนิวรอนในชั้นเอาต์พุตของโครงข่ายประสาทที่ฝึกเป็นตัวควบคุม โดยอาศัยการส่งผ่านทางชั้นต่างๆจนถึงชั้นอินพุตของโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ ค่าความคลาดเคลื่อนจะถูกนำมาใช้ในการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมโดยมีเป้าหมายเพื่อลดค่าความคลาดเคลื่อนดังกล่าว ดังแสดงในรูปที่ 7



ภาพที่ 24 การฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุม

การฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมเป็นการฝึกแบบ off-line จะไม่มีการรวมพลานต์จริงเข้ามาในระหว่างการฝึก แต่อาศัยโครงข่ายประสาทในการหาเอกลักษณ์แทนพลานต์จริง

เมื่อป้อนข้อมูลอินพุต (U) โครงข่ายประสาทที่ต้องการเป็นตัวควบคุม สามารถแสดงดังแสดงสมการที่ 31

$$U_c(k+1) = f_{NN_c}(U(k), ref(k)) \quad (31)$$

โดยที่

$f_{NN_c}\{\bullet\}$	โครงข่ายประสาทที่ใช้เป็นตัวควบคุม
$(U(k), ref(k))$	ชุดข้อมูลอินพุตที่ป้อนเข้าสู่โครงข่ายที่ใช้เป็นตัวควบคุม
$U_c(k+1)$	สัญญาณควบคุม
$U(k)$	สัญญาณอินพุต
$ref(k)$	สัญญาณอ้างอิง

ชุดข้อมูลอินพุต ($U(k), ref(k)$) ซึ่งป้อนเข้าสู่โครงข่ายประสาทที่ใช้เป็นตัวควบคุม จะถูกส่งผ่านไปยังชั้นต่างๆจนถึงชั้นเอาต์พุตของโครงข่ายประสาท สัญญาณออกจากป็นิรอนในชั้นดังกล่าวจะเป็นสัญญาณควบคุม ซึ่งถูกป้อนเข้าสู่โครงข่ายประสาทหลังจากได้รับการฝึกในการหาเอกลักษณ์ แสดงความสัมพันธ์ดังสมการที่ 32

$$Y_m(k+1) = fNN_m(U_c(k), ref(k)) \quad (32)$$

ค่าความคลาดเคลื่อนระหว่างสัญญาณออกจากโครงข่ายประสาทที่ได้รับการฝึกในการหาเอกลักษณ์ (Y_m) กับสัญญาณอ้างอิง: E_2 นำไปใช้ในการกำหนดค่าบรรณของการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุม ดังแสดงในสมการที่ 33

$$JNN_c = \frac{1}{2} \sum_{k=0}^N (ref(k) - Y_m(k))^2 \quad (33)$$

ค่าความคลาดเคลื่อน E_2 ถูกถ่ายทอดผ่านทุกชั้นของโครงข่ายประสาทที่ได้รับการฝึกในการหาเอกลักษณ์ พิจารณาในทำนองเดียวกับขั้นตอนการประมวลผลของโครงข่ายประสาทเทียมดังที่กล่าวมาข้างต้น โดยกำหนดค่าถ่วงน้ำหนัก และค่าไบแอสโครงข่ายประสาทเทียมที่ได้รับการฝึกในการหาเอกลักษณ์เป็นค่าคงที่ และไม่มีการเปลี่ยนแปลงตลอดการฝึกโครงข่ายประสาทเทียมเพื่อเป็นตัวควบคุม ดังแสดงในสมการต่อไปนี้

สำหรับชั้นเอาต์พุต

$$\delta_o = [T(k) - Y(k)] f'(y_{in}) \quad (34)$$

สำหรับชั้นภายใน

$$\delta_h = (W\delta_o) f'(Z_{in}) \quad (35)$$

สำหรับชั้นอินพุต

$$E_0 = V\delta_h \quad (36)$$

สัญญาณความคลาดเคลื่อนของปมนิวรอนในชั้นอินพุต (E_0) จะคำนวณได้จากสมการที่ 36 ใช้ในการปรับค่าน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมเพื่อเป็นตัวควบคุม การใช้งานโครงข่ายประสาทเทียมหลักจากได้รับการฝึกในการหาเอกลักษณ์และเพื่อเป็นตัวควบคุม

การใช้งานโครงข่ายประสาทเทียมในระบบควบคุมจริง ดังแสดงในรูปที่ 5 อาศัยค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมหลังการฝึกทั้งในส่วนการหาเอกลักษณ์และในส่วนของตัวควบคุม เป็นจุดเริ่มต้นการทำงาน ซึ่งค่าพารามิเตอร์ต่างๆของโครงข่ายประสาทเทียมทั้งสองจะถูกปรับให้สอดคล้องกับการเปลี่ยนแปลงที่เกิดขึ้นกับระบบเพื่อให้สมรรถนะโดยรวมของระบบเป็นไปตามต้องการ

การทำงานของระบบควบคุมที่ใช้โครงข่ายประสาทเทียม เริ่มจากการกำหนดกรณีสมรรถนะของโครงข่ายประสาทเทียมที่ใช้เป็นตัวควบคุม ดังแสดงในสมการที่ 37

$$JNN_c = \frac{1}{2} (ref(k) - Y(k))^2 \quad (37)$$

โครงข่ายประสาทเทียมที่ใช้เป็นตัวควบคุมจะรับสัญญาณเข้า $U(k)$ และสร้างสัญญาณควบคุม U_c ดังแสดงในสมการที่ 38

$$U_c(k+1) = fNN_c(U(k), ref(k)) \quad (38)$$

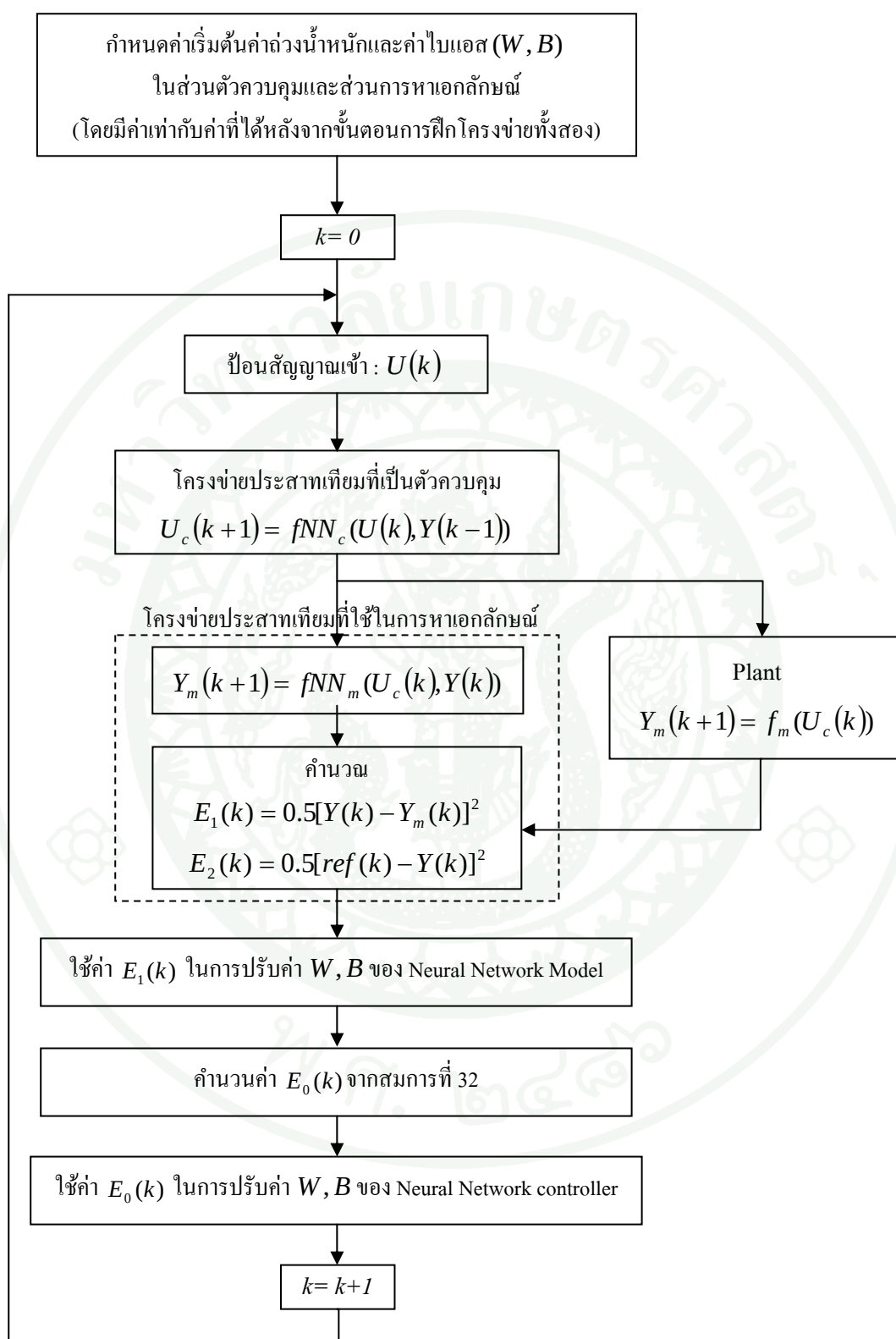
สัญญาณควบคุมป้อนเข้าสู่ผู้พลานท์และโครงข่ายประสาทเทียมที่ใช้ในการหาเอกลักษณ์ สัญญาณที่ออกคำนวณได้ดังสมการที่ 39

$$Y_m(k+1) = fNN_m(U_c(k), Y(k)) \quad (39)$$

โดยกรณีสมรรถนะของโครงข่ายประสาทเทียมที่ใช้ในการหาเอกลักษณ์ แสดงดังสมการที่ 40

$$JNN_c = \frac{1}{2} (Y(k) - Y_m(k))^2 \quad (40)$$

การปรับค่าพารามิเตอร์ของโครงข่ายประสาทเทียมทั้งสอง พิจารณาในทำนองเดียวกับขั้นตอนการประมวลผลของโครงข่ายประสาทเทียม สามารถสรุปได้ดังรูปที่ 8



ภาพที่ 25 แผนภาพของระบบควบคุมโดยใช้โครงข่ายประสาทเทียม

งานวิจัยที่เกี่ยวข้อง

กิดชอบ (2544) ได้ทำการพัฒนาระบบวัดแรงบิด และความเร็วสำหรับการทดสอบ เครื่องจักรกลไฟฟ้า โดยอาศัยทฤษฎีของการควบคุมมอเตอร์แบบปราศจากเซนเซอร์วัดความเร็ว และแรงบิด ซึ่งจะทำการตรวจจับแรงดันและกระแสของมอเตอร์ มาคำนวณค่าแรงบิดและความเร็วด้วยตัวประมวลผลเชิงดิจิทัล แทนการใช้เซนเซอร์วัดแรงบิดและความเร็วของมอเตอร์ โดยตรง ผลการทดสอบระบบที่พัฒนาขึ้นแสดงให้เห็นว่า ระบบสามารถวัดหรือควบคุมแรงบิดและความเร็วได้อย่างถูกต้อง โดยมีค่าผิดพลาดไม่เกิน $+10\%$ และ $+1\%$ ของค่าพิกัดตามลำดับ

รดาณูช (2544) ได้ใช้โครงข่ายประสาทในการประมาณค่า ออกแบบแบบจำลองและตัวควบคุมสำหรับกระบวนการย่อยเยื่อกระดาษโดยโครงข่ายประสาทแบบป้อนไปข้างหน้าหลายชั้น และแบบกระจายความผิดพลาดย้อนกลับในการฝึกข่ายงาน ผลคือเมื่อใช้ตัวควบคุมโครงข่ายประสาทสามารถควบคุมให้สู่เป้าหมายในเวลาที่กำหนดและมีสมรรถนะที่ดี

วิศณีย์ (2544) ได้สร้างแบบจำลองฟัซซี่โดยใช้ข้อมูลอินพุต-เอาต์พุตของกระบวนการผ่านวิธีคลัสเตอร์และวิธีลีสท์สแควร์ เพื่อใช้เป็นแบบจำลองภายในสำหรับการควบคุมแบบอาศัยแบบจำลองภายในของตัวย่อยเยื่อกระดาษ ผลคือตัวประมาณค่าฟัซซี่สามารถประมาณค่าได้อย่างมีประสิทธิภาพ สำหรับตัวควบคุมฟัซซี่ที่ใช้ร่วมกับตัวประมาณค่าฟัซซี่ สามารถควบคุมค่าสุดท้ายให้เข้าใกล้เป้าหมายได้ในเวลาที่กำหนด

วฤต (2534) ได้ทำการพิจารณาปัญหาตัวควบคุมตัวเดียวกันสำหรับระบบเชิงเส้นหลายระบบ โดยพิจารณาสมรรถนะในเชิง H2 พร้อมทั้งได้เสนอวิธีการหาตัวควบคุม โดยแบ่งเป็น 2 ขั้นตอน ได้แก่ ขั้นตอนหาเมทริกซ์ป้อนกลับสถานะร่วม และขั้นตอนหาเมทริกซ์สังเกตสถานะร่วม ซึ่งแต่ละขั้นตอนใช้สมการเมทริกซ์เชิงเส้นเป็นเครื่องมือในการหาคำตอบ

Ahn and Chau (2007) นำ QFT ทำการออกแบบตัวควบคุมสำหรับระบบ Hybrid actuation เพื่อศึกษาการควบคุมแรงในระบบ โดยสร้างขอบเขตสำหรับการหลีกเลี่ยงสัญญาณรบกวนภายนอกของระบบ (Plant-output disturbance rejection) และขอบเขตสำหรับการติดตามการเคลื่อนที่ (Tracking) นำผลการทดลองมาเปรียบเทียบกับกรออกแบบตัวควบคุมแบบ PID

Akhyar and Omatu เสนอโครงสร้างของตัวควบคุม PID แบบปรับตัวเอง (Self-Tuning PID Controller) ซึ่งนำเทคนิคโครงข่ายประสาทเทียมมาสร้างแบบจำลองของกระบวนการ เพื่อใช้ในการปรับจูนตัวควบคุม PID จากผลการจำลองด้วยคอมพิวเตอร์พบว่า จากโครงสร้างของตัวควบคุม PID ที่ใช้ สามารถควบคุมให้ผลตอบสนองของระบบเชิงเส้นและระบบไม่เชิงเส้นเปลี่ยนแปลงตามค่าจุดตั้งปรับได้อย่างรวดเร็ว

Chen ได้นำเสนอเทคนิคโครงข่ายประสาทเทียมในโครงสร้างของตัวควบคุมแบบปรับตัวเอง (Self Tuning Adaptive Controller) กับระบบที่ไม่เป็นเชิงเส้นซึ่งเขียนอยู่ในรูปของสมการไม่ต่อเนื่องและไม่ทราบค่าพารามิเตอร์ของระบบ (Unknown discrete-time nonlinear system)

Cui and Shin ได้ทดสอบโครงข่ายประสาทเทียมในการควบคุมแบบ Direct Adaptive Control ระบบควบคุมที่มีส่วนไม่เป็นเชิงเส้น ซึ่งได้แก่ เขตไร้ผลสนอง (Dead Zone) และระบบที่มีการประวิงเวลา (Time Delay)

Irwin and Lightbody ได้เสนอตัวควบคุมปรับตัวเองแบบจำลองในการควบคุมถึงปฏิกรณ์เคมี ในการเปรียบเทียบผลการควบคุมกับตัวควบคุม PI พบว่าผลตอบสนองที่ได้สามารถตามรอยสัญญาณอ้างอิงได้ดีกว่าตัวควบคุม PI มาก

Jin, Nikiforuk and Gupta ได้ประยุกต์ใช้ Recurrent Neural Network ในการประมาณพลวัตของระบบซึ่งอยู่ในรูปของสมการที่ไม่ต่อเนื่องทางเวลา (Discrete Time System) กับการควบคุมแบบ MRAC ซึ่งมีศักยภาพดีกว่าโครงข่ายประสาทเทียมแบบเก่า

Kundergi and Nataraj (1994) ได้นำ QFT มาศึกษาในการออกแบบการควบคุมการทำงานของ Evaporator โดยมีลักษณะของระบบเป็นแบบ 2×2 และมีความไม่แน่นอนคือ ค่า Time-delays และ Gains

Leuba and Perez (1988) ได้ออกแบบตัวควบคุมโดยใช้เทคนิค Optimal control ร่วมกับ Pontryagin's maximum ในการติดตามเทอมที่ไม่ทราบค่าของระบบ

Narendra and Parthasarathy ได้นำเสนอเทคนิคโครงข่ายประสาทเทียมในการควบคุมแบบ Model Reference Adaptive Control (MRAC) และทดสอบผลโดยวิธี Simulation กับระบบที่เป็นเชิงเส้นและไม่เป็นเชิงเส้น

Nordgren and Meckl ได้ทำการวิเคราะห์และเปรียบเทียบความเหมือนและแตกต่างของกฎการเรียนรู้และกฎการปรับเปลี่ยน (Learning/Adaptive Law) และสมรรถนะของระบบ รวมทั้งเทคนิคการวิเคราะห์เสถียรภาพของ Lyapunov ซึ่งนำมาใช้ใน Adaptive Law ของโครงข่ายประสาทเทียมที่นำมาใช้ควบคุมระหว่าง Neural Network Inverse Dynamic Controller กับ Model-Based Adaptive Controller

Sastry, Santharam and Unnikrishnan ได้ประยุกต์ใช้ Memory Neural Network ซึ่งเป็นโครงข่ายประสาทเทียมที่มีโครงสร้างผสมระหว่าง Recurrent Neural Network กับ Feedforward Neural Network ในระบบควบคุมแบบ MRAC โดยใช้วิธี Differentiating the model หรือ Forward and inverse modeling ในการปรับตัวควบคุม

Yamamura (1990) ได้นำเสนอเทคนิค Neural network ในการควบคุมแขนกลชนิดสองข้อต่อ โดยใช้ Error back propagation (BEP) และ Modified versions of BEP

Yang and Linkens ได้ใช้ตัวควบคุมแบบปรับตัวเองในเวลาจริงที่อาศัยโครงข่ายประสาทเทียม (Online Adaptive Neural Network Controller) ควบคุมถังปฏิกรณ์เคมี (Stirred Tank Reactor) ซึ่งใช้เป็นตัวแทนของระบบไม่เชิงเส้น

อุปกรณ์และวิธีการ

อุปกรณ์

1. เครื่องขันสกรู (Screwdriver)
2. อุปกรณ์วัดแรงบิด (Torque Transducer)
3. โปรแกรมคอมพิวเตอร์แมทแลป (Matlab R2010a)
4. คอมพิวเตอร์พกพา (Computer Notebook)

วิธีการ

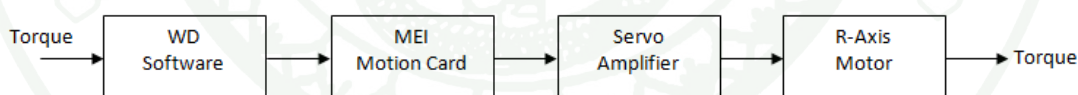
เนื่องจากการออกแบบระบบควบคุมโดยโครงข่ายประสาทเทียมมีความแตกต่างจากการออกแบบตัวควบคุมแบบอื่น โดยโครงข่ายประสาทเทียมจะประมาณหน้าที่ของอุปกรณ์โดยเฉพาะเมื่อเราไม่ทราบแบบจำลองทางคณิตศาสตร์ของแพลนท์ โครงข่ายประสาทจะทำให้สามารถออกแบบตัวควบคุมได้ โดยเราต้องมีข้อมูลเกี่ยวกับระบบในรูปแบบของข้อมูลป้อนเข้า (Input) ผลลัพธ์ (Output) ดังนั้นตัวควบคุมโครงข่ายประสาทจะได้จากการเรียนรู้โครงข่ายประสาท และความสามารถที่เหมาะสมสำหรับการปรับตัวควบคุมต้องปรับให้เข้ากับการเปลี่ยนแปลงสภาพแวดล้อม โดยอาศัยรูปแบบจากกลุ่มตัวอย่างอินพุต เอาต์พุต ที่ใช้ในการปรับสอน ซึ่งความสามารถเหล่านี้เรียกว่า การเรียนรู้จากประสบการณ์ (Learn from Experience) งานวิจัยนี้จึงเลือกใช้โครงข่ายประสาทเทียมมาใช้ในการหาความสัมพันธ์ของแรงบิด (Target) กับความเร็ว (Input) ที่ป้อนเข้าระบบเพื่อใช้ในการหาเอกลักษณ์ และออกแบบระบบควบคุมของเครื่องขันสกรูอัตโนมัติ โดยขั้นตอนของการวิจัยสามารถแบ่งออกเป็น 7 ส่วนได้ดังนี้คือ 1. การกำหนดวัตถุประสงค์และขอบเขต 2. การศึกษากระบวนการขันสกรูและการทำงานของเครื่องขันสกรูอัตโนมัติ 3. การศึกษาทฤษฎีของโครงข่ายประสาทเทียม การศึกษาการเขียนโปรแกรมคอมพิวเตอร์แมทแลป 4. กำหนดตัวแปรป้อนเข้าและตัวเป้าหมายที่ใช้ในการหาเอกลักษณ์ 5. การหาเอกลักษณ์ของระบบและการออกแบบระบบควบคุมโครงข่ายประสาท 6. การจำลองการทำงานและการวิเคราะห์ผลของระบบควบคุมโดยโครงข่ายประสาท 7. การสรุปและเสนอแนะ โดยแสดงขั้นตอนการดำเนินงานวิจัยในรูปที่ 26

1. การกำหนดวัตถุประสงค์และขอบเขต

งานวิจัยนี้มีวัตถุประสงค์ในการนำเสนอหลักการ และทฤษฎีของโครงข่ายประสาทเทียม มาประยุกต์ใช้ในการหาเอกลักษณ์ และการออกแบบระบบควบคุมกระบวนการขึ้นสกรูของ เครื่องขึ้นสกรูอัตโนมัติ โดยการเขียนโปรแกรมคอมพิวเตอร์แมทแลป และออกแบบโครงข่าย ประสาทเทียมที่เหมาะสมที่ทำให้ค่าความคลาดเคลื่อนต่ำที่สุด เพื่อหาเอกลักษณ์และระบบควบคุม กระบวนการขึ้นสกรูของเครื่องขึ้นสกรูอัตโนมัติ

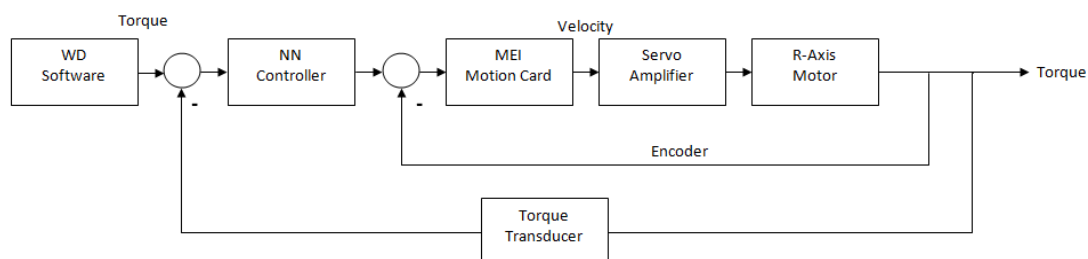
2. การศึกษากระบวนการขึ้นสกรูและการทำงานของเครื่องขึ้นสกรูอัตโนมัติ

งานวิจัยนี้ได้ศึกษาการขึ้นสกรูเพื่อประกอบกล่องปกคลุมฮาร์ดดิสก์ (Top Cover Case) ใน ระบบควบคุมกระบวนการขึ้นสกรูในปัจจุบันนั้น เอซีเซอร์โวมอเตอร์ (AC Servo Motor) จะถูก ควบคุมการขึ้นสกรูแต่ละตัวด้วยการกำหนดตำแหน่งความลึกของตัวสกรู แสดงในรูปที่ 26 โดยที่ ระยะความลึกของสกรูจะถูกควบคุมโดยเอซีเซอร์โวมอเตอร์แกน R (R-Axis AC Servo Motor) ความลึกสุดท้ายของการขึ้นสกรูถูกส่งกลับไปที่ตัวควบคุมเพื่อเป็นการยืนยันตำแหน่ง ผลลัพธ์ แรงบิดของการขึ้นสกรูจะไม่ถูกส่งให้ระบบควบคุมโดยตรง จึงทำให้ระบบควบคุมไม่รู้ค่าแรงบิด ที่แท้จริงของการขึ้นสกรูในระหว่างกระบวนการขึ้น



ภาพที่ 26 ระบบควบคุมกระบวนการขึ้นสกรูปัจจุบัน

ระบบควบคุมกระบวนการขึ้นสกรูที่จะออกแบบในงานวิจัยนี้จะเป็นการควบคุมแรงบิด ของการขึ้นสกรูโดยตรง โดยจะมีอุปกรณ์วัดแรงบิด (Transducer Torque) ถูกติดตั้งระหว่าง เอซีเซอร์โวมอเตอร์กับหัวขึ้นสกรู เพื่อเป็นตัววัดค่าแรงบิดโดยตรงระหว่างกระบวนการขึ้นสกรู แสดงในภาพที่ 27

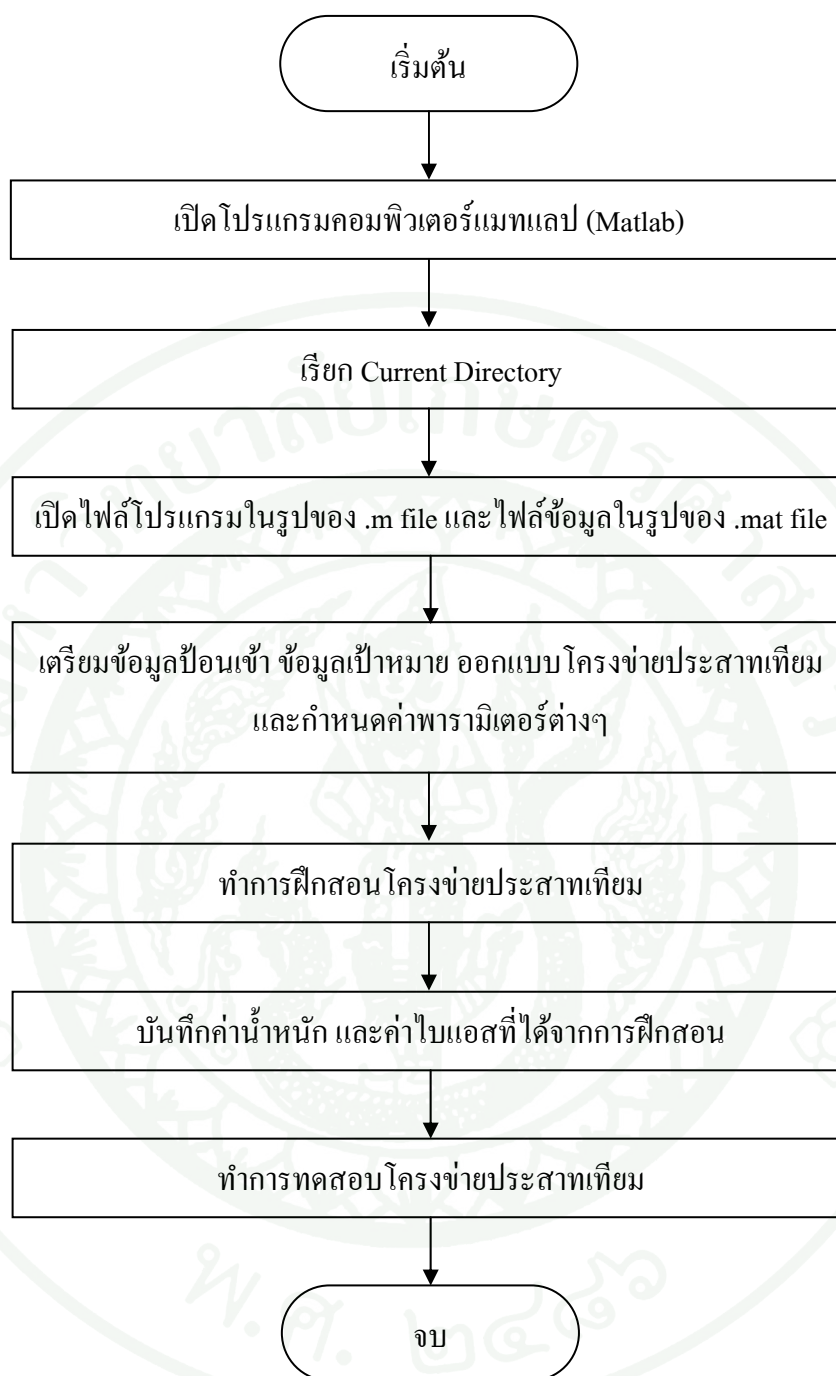


ภาพที่ 27 ระบบควบคุมที่ใช้ในงานวิจัยนี้

3. การศึกษาทฤษฎีของโครงข่ายประสาทเทียม การศึกษาการเขียนโปรแกรมคอมพิวเตอร์แมทแลป

3.1 งานวิจัยนี้จะศึกษาทฤษฎีและรูปแบบการใช้งานของโครงข่ายประสาทเทียม ซึ่งการหาโครงข่ายประสาทเทียมที่เหมาะสมที่ทำให้ค่าความคลาดเคลื่อนในการหาเอกลักษณ์ และระบบควบคุมต่ำที่สุดนั้น จะใช้โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ (Backpropagation Neural Network) เพื่อปรับเปลี่ยนค่าน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมและเพื่อปรับผลลัพธ์ของแบบจำลองให้ใกล้เคียงกับค่าความจริงมากที่สุด

3.2 ศึกษาการเขียนโปรแกรมคอมพิวเตอร์แมทแลป (Matlab) เพื่อใช้ในการดำเนินงาน โดยขั้นตอนการใช้งานโครงข่ายประสาทเทียมในโปรแกรมคอมพิวเตอร์แมทแลป แสดงดังภาพที่ 28 และการทำงานของโครงข่ายประสาทเทียมในโปรแกรมคอมพิวเตอร์แมทแลป แสดงดังภาพผนวก ก ภาพผนวก ข และภาพผนวก ค



ภาพที่ 28 แสดงขั้นตอนการใช้งานโครงข่ายประสาทเทียมในโปรแกรมแมทแลป

4. กำหนดตัวแปรป้อนเข้าและตัวเป้าหมายที่ใช้ในการหาเอกลักษณ์โครงข่ายประสาทเทียม

งานวิจัยนี้ได้ใช้โครงสร้างของโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับในการเรียนรู้ โดยที่โครงข่ายประสาทเทียมนั้นประกอบด้วยชั้นรับข้อมูล ชั้นซ่อนและแสดงผล โดยชั้นรับข้อมูลจะเป็นตัวแปรป้อนเข้า คือ ข้อมูลของค่าความเร็วที่ส่งให้กับเอชชีเซอร์ไวโมเตอร์ ส่วนตัวเป้าหมายจะเป็นค่าแรงบิดที่ได้ระหว่างกระบวนการขับเคลื่อน

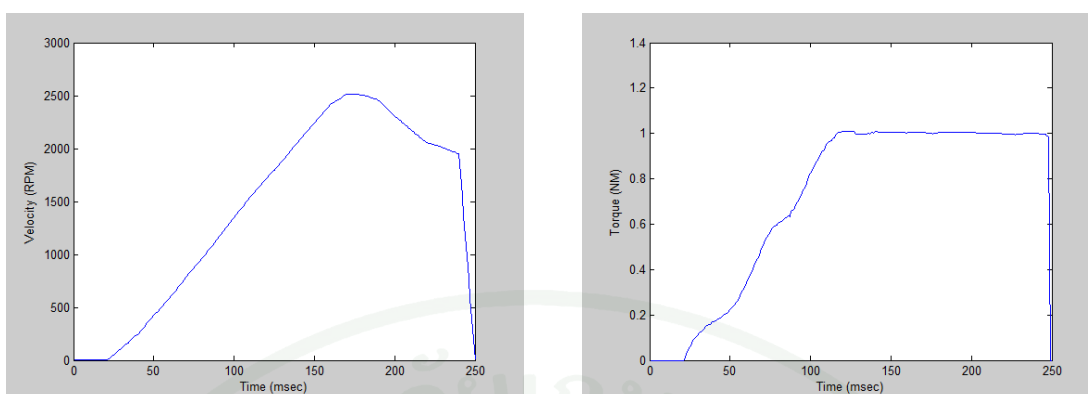
5. การหาเอกลักษณ์ของระบบและการออกแบบระบบควบคุมโครงข่ายประสาท

ขั้นตอนการเรียนรู้ เริ่มจากการกำหนดโครงสร้างของโครงข่ายประสาทเทียม ได้แก่ ฟังก์ชันการถ่ายโอน (Transfer Function) จำนวนชั้น (Number of Network Layers) ของโครงข่ายประสาทเทียม จำนวนข้อมูลป้อนเข้า (Number of Input Nodes) จำนวนนิวรอนของชั้นซ่อนและชั้น (Hidden Layer Nodes) และจำนวนผลลัพธ์ (Number of Output Nodes) จากนั้นนำข้อมูลป้อนเข้า และข้อมูลเป้าหมายมาป้อนให้กับโครงสร้างของโครงข่ายประสาทเทียม เพื่อที่จะคำนวณหาความสัมพันธ์ระหว่างอินพุตกับเอาต์พุต โดยค่าความสัมพันธ์นี้จะอยู่ในรูปของค่าน้ำหนัก และค่าไบแอส หลังจากนั้นจะนำค่าเอาต์พุตที่ได้มาเปรียบเทียบกับค่าเป้าหมาย เพื่อคำนวณหาความคลาดเคลื่อนทั้งหมดที่เกิดขึ้น ถ้าค่าความคลาดเคลื่อนทั้งหมดมากกว่าค่าที่กำหนด โปรแกรมจะทำการปรับค่าน้ำหนักและค่าไบแอส แล้วคำนวณค่าความสัมพันธ์ใหม่ทั้งหมดอีกครั้ง ขั้นตอนการเรียนรู้นี้แสดงดังภาพที่ 19 และเมื่อค่าความคลาดเคลื่อนทั้งหมดเท่ากับ หรือน้อยกว่าค่าที่กำหนด โครงข่ายจะหยุดการเรียนรู้และทำการคำนวณค่าประสิทธิภาพของโครงข่ายประสาทเทียม และทำการบันทึกค่าน้ำหนักและค่าไบแอสเพื่อนำไปใช้ในกระบวนการทดสอบต่อไป

5.1 โครงข่ายประสาทเทียมในการหาเอกลักษณ์

5.1.1 เตรียมชุดข้อมูลที่ใช้ในการฝึกโครงข่ายประสาท

ชุดข้อมูลที่นำมาใช้ในการทดสอบประกอบด้วย ชุดข้อมูลอินพุตและชุดข้อมูลเป้าหมาย ซึ่งชุดข้อมูลอินพุตเป็นค่าที่ป้อนเข้าสู่พลานท์ คือ ค่าความเร็ว ส่วนชุดเป้าหมายเป็นค่าที่ออกจากพลานท์ คือ ค่าทอร์ค จำนวน 250 ชุด คาบเวลาชักตัวอย่างเท่ากับ 1 มิลลิวินาที ตัวอย่างชุดข้อมูลแสดงในภาพที่ 29



(a) Velocity Command

(b) Torque Signal

ภาพที่ 29 ชุดข้อมูลอินพุตที่ใช้ฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์

5.1.2 ฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์

การกำหนดลักษณะโครงสร้างของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ เนื่องจากไม่มีกฎเกณฑ์ที่แน่นอนในการกำหนดลักษณะโครงสร้าง ดังนั้นจึงใช้วิธีการลองผิดลองถูก โดยเริ่มจากลักษณะโครงสร้างที่ซับซ้อนน้อยที่สุดและเพิ่มขึ้นเรื่อยๆ ทั้งนี้ต้องคำนึงถึงความสามารถในการนำไปใช้งานกับระบบควบคุมจริงให้เป็นไปได้อย่างมีประสิทธิภาพ ลักษณะโครงสร้างของโครงข่ายประสาทที่ใช้ในการฝึกมี จำนวน 3 ชั้น ได้แก่ชั้นอินพุต ชั้นซ่อน และชั้นเอาต์พุต โดยจำนวนโหนดในชั้นอินพุตและชั้นเอาต์พุตเท่ากับจำนวนสัญญาณเข้าและออก โครงข่ายประสาท ซึ่งมี จำนวน 2 และ 1 โหนด ตามลำดับ โดยเปลี่ยนจำนวนโหนดในชั้นซ่อน เพื่อหาโครงสร้างของโครงข่ายประสาทที่เหมาะสมที่สุดในการหาเอกลักษณ์ ซึ่งในการฝึกนั้น จำนวนโหนดในชั้นซ่อนจะใช้ตั้งแต่ 10 โหนด จนถึง 200 โหนด โดยเพิ่มทีละ 10 โหนด และเปรียบเทียบประสิทธิภาพของโครงข่ายประสาทเทียม ว่าโครงสร้างไหนมีประสิทธิภาพมากที่สุด แสดงผลดังตารางที่ 1

ตารางที่ 1 แสดงผลการฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์

โครงสร้างโครงข่าย		อัตราความสำเร็จ
1.	$2 \times 10 \times 1$	3.16
2.	$2 \times 20 \times 1$	2.12
3.	$2 \times 30 \times 1$	2.13
4.	$2 \times 40 \times 1$	1.07
5.	$2 \times 50 \times 1$	1.11
6.	$2 \times 60 \times 1$	0.80
7.	$2 \times 70 \times 1$	0.62
8.	$2 \times 80 \times 1$	0.52
9.	$2 \times 90 \times 1$	0.48
10.	$2 \times 100 \times 1$	0.52
11.	$2 \times 110 \times 1$	0.49
12.	$2 \times 120 \times 1$	0.42
13.	$2 \times 130 \times 1$	0.43
14.	$2 \times 140 \times 1$	0.41
15.	$2 \times 150 \times 1$	0.40
16.	$2 \times 160 \times 1$	0.40
17.	$2 \times 170 \times 1$	0.50
18.	$2 \times 180 \times 1$	0.40
19.	$2 \times 190 \times 1$	0.40
20.	$2 \times 200 \times 1$	0.41

ส่วนฟังก์ชันการถ่ายโอนที่ใช้ในโครงข่ายประสาทนี้คือ ฟังก์ชันการถ่ายโอนแบบลอจิสติกมอยต์ในชั้นซ่อน และฟังก์ชันการถ่ายโอนแบบลิเนียร์ในชั้นเอาต์พุต จำนวนชุดข้อมูลในการฝึก 249 ชุด จำนวนรอบการเรียนรู้ 3000 รอบ โดยผลการฝึกโครงข่ายประสาทเทียม แสดงในภาพที่ 38

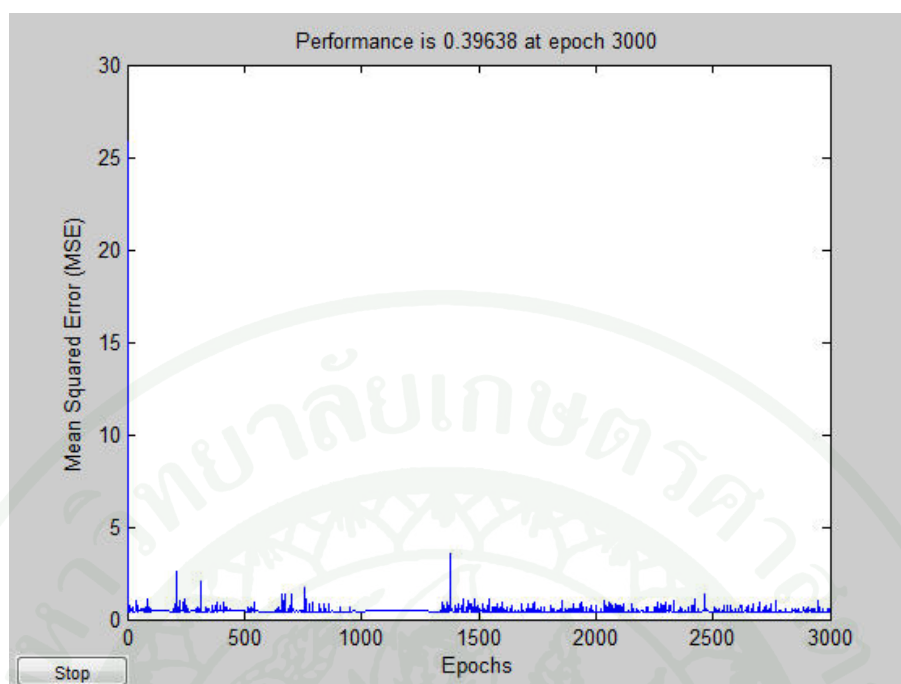
จากตารางที่ 1 เพื่อให้การเรียนรู้ของโครงข่ายประสาทเทียมในการหาเอกลักษณ์มีประสิทธิภาพสูง จึงเลือกโครงสร้างที่มีจำนวนชั้นซ่อนเท่ากับ 150 โหนด ซึ่งมีค่าอัตรา

สมรรถนะของโครงข่ายประสาทเทียมประมาณ 0.40 เนื่องจากเมื่อเพิ่มจำนวนโหนดเข้าไปแล้ว
 ครรชนีสมรรถนะของการฝึกโครงข่ายประสาทเทียมมีค่าใกล้เคียงกัน ดังนั้นเพื่อให้การเรียนรู้ของ
 โครงข่ายประสาทเทียมมีประสิทธิภาพดังเลือกลักษณะโครงข่ายนี้ โดยมีโครงสร้างของโครงข่าย
 ดังแสดงในตารางที่ 2

ตารางที่ 2 แสดงลักษณะ โครงสร้างของโครงข่ายประสาทเทียมในการหาเอกลักษณ์

โครงข่ายประสาทเทียมในการหาเอกลักษณ์	โครงสร้าง
จำนวนชั้น	3
จำนวนโหนดในชั้นอินพุต	2
จำนวนโหนดในชั้นซ่อน	150
จำนวนโหนดในชั้นเอาต์พุต	1
ฟังก์ชันการถ่ายโอนในชั้นซ่อน	$a = 1/(1+e^{-n})$
ฟังก์ชันการถ่ายโอนในชั้นเอาต์พุต	$a = n$
อัตราการเรียนรู้	0.01

ครรชนีสมรรถนะขณะเรียนรู้แสดงดังภาพที่ 30 จากนั้นนำโครงสร้างโครงข่ายประสาท
 เทียมเอกลักษณ์นี้ไปใช้ในการเรียนรู้โครงข่ายประสาทเทียมในการควบคุมต่อไป

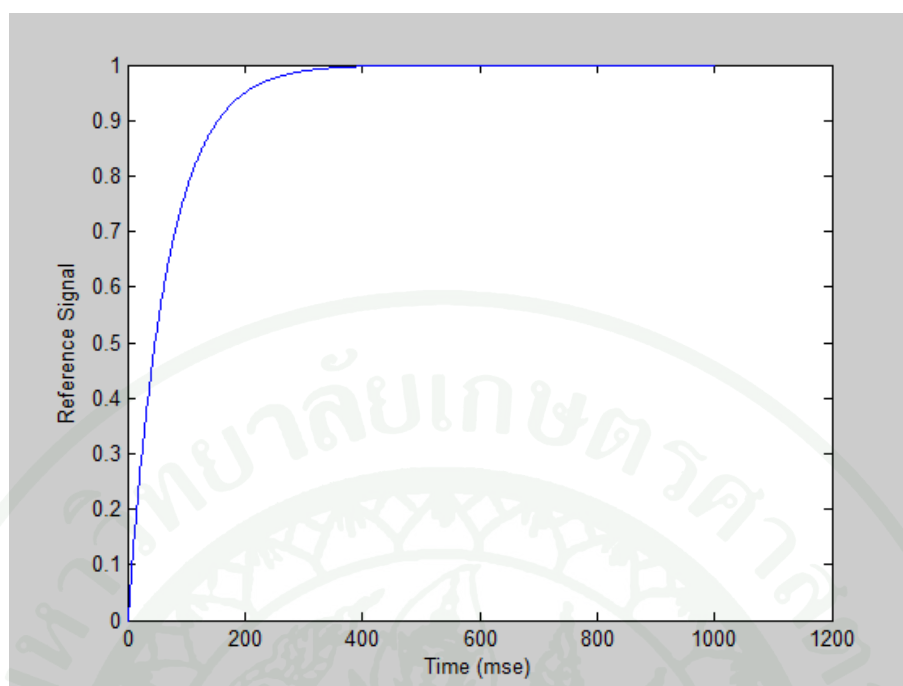


ภาพที่ 30 ครรชนีสมรรถนะขณะเรียนรู้ของของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่จำนวนโหนด เท่ากับ 150

5.2 โครงข่ายประสาทเทียมในการควบคุม

5.2.1 เตรียมชุดข้อมูลที่ใช้ในการฝึกโครงข่ายประสาท

ชุดข้อมูลสำหรับการทดสอบโครงข่ายประสาทเทียมในการควบคุมนั้น จะใช้สัญญาณอ้างอิงเป็นชุดข้อมูลอินพุตและข้อมูลเป้าหมาย แสดงในภาพที่ 31



ภาพที่ 31 ชุดข้อมูลสำหรับใช้ฝึกโครงข่ายประสาทเทียมในการควบคุม

5.2.2 ฝึกโครงข่ายประสาทเทียมในการควบคุม

การกำหนดลักษณะ โครงสร้างของโครงข่ายประสาทเทียมในการควบคุมเนื่องจากไม่มีกฎเกณฑ์ที่แน่นอนในการกำหนดลักษณะโครงสร้าง ดังนั้นจึงใช้วิธีการลองผิดลองถูก โดยเริ่มจากลักษณะโครงสร้างที่ซับซ้อนน้อยที่สุดและเพิ่มขึ้นเรื่อยๆ ทั้งนี้ต้องคำนึงถึงความสามารถในการนำไปใช้งานกับระบบควบคุมจริงให้เป็นไปได้อย่างมีประสิทธิภาพ ลักษณะโครงสร้างของโครงข่ายประสาทที่ใช้ในการฝึกควบคุมมีจำนวน 3 ชั้น เช่นเดียวกับการฝึกการหาเอกลักษณ์ โดยจำนวนโหนดในชั้นอินพุตและชั้นเอาต์พุตเท่ากับจำนวนสัญญาณเข้าและออกโครงข่ายประสาท ซึ่งมี จำนวน 2 และ 1 โหนด ตามลำดับ โดยเปลี่ยนจำนวนโหนดในชั้นซ่อน เพื่อหาโครงสร้างของโครงข่ายประสาทที่เหมาะสมที่สุดในการหาเอกลักษณ์ ซึ่งในการฝึกนั้นจำนวนโหนดในชั้นซ่อนจะใช้ตั้งแต่ 5 โหนด จนถึง 70 โหนด โดยเพิ่มทีละ 5 โหนด และเปรียบเทียบประสิทธิภาพของโครงข่ายประสาทเทียม ว่าโครงสร้างไหนมีประสิทธิภาพมากที่สุด แสดงผลดังตารางที่ 3

ตารางที่ 3 แสดงผลการฝึกโครงข่ายประสาทเทียมในการหาเอกลักษณ์

โครงสร้างโครงข่าย		อัตราความสำเร็จ
1.	$2 \times 5 \times 1$	0.0386
2.	$2 \times 10 \times 1$	0.0443
3.	$2 \times 15 \times 1$	0.0440
4.	$2 \times 20 \times 1$	0.1514
5.	$2 \times 25 \times 1$	0.1269
6.	$2 \times 30 \times 1$	0.0183
7.	$2 \times 35 \times 1$	0.0926
8.	$2 \times 40 \times 1$	0.0256
9.	$2 \times 45 \times 1$	0.0245
10.	$2 \times 50 \times 1$	0.0159
11.	$2 \times 55 \times 1$	0.0580
12.	$2 \times 60 \times 1$	0.0148
13.	$2 \times 65 \times 1$	0.0364
14.	$2 \times 70 \times 1$	0.0586

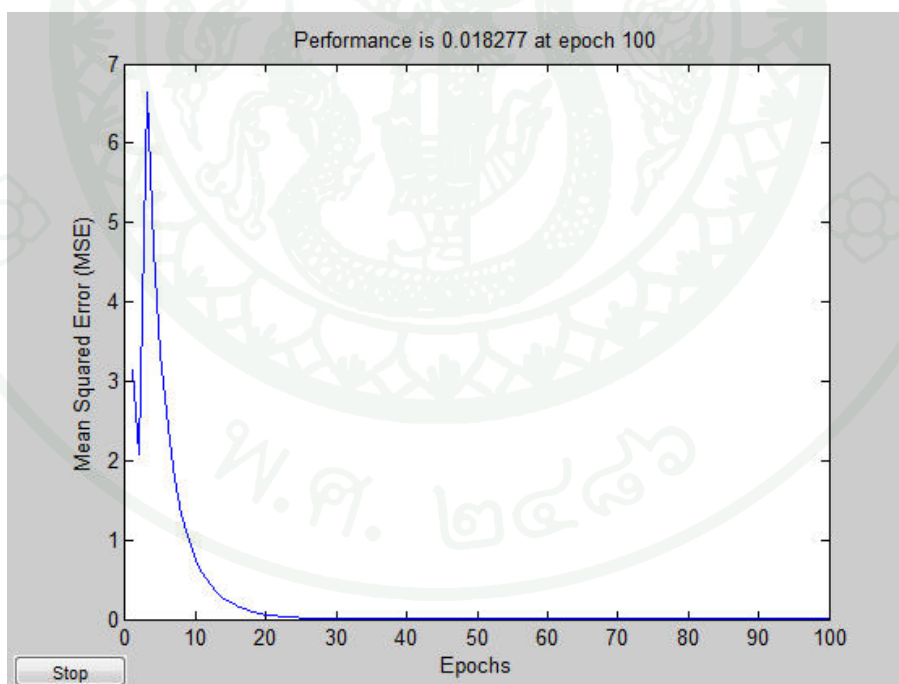
ส่วนฟังก์ชันการถ่ายโอนที่ใช้ในโครงข่ายประสาทนี้คือ ฟังก์ชันการถ่ายโอนแบบลอจิสติกมอดในชั้นซ่อน และฟังก์ชันการถ่ายโอนแบบลิเนียร์ในชั้นเอาต์พุต จำนวนชุดข้อมูลในการฝึก 1000 ชุด จำนวนรอบการเรียนรู้ 100 รอบ โดยผลการฝึกโครงข่ายประสาทเทียม แสดงในภาพที่ 40

จากตารางที่ 3 เพื่อให้การเรียนรู้ของโครงข่ายประสาทเทียมในการควบคุมมีประสิทธิภาพสูง จึงเลือกโครงสร้างที่มีจำนวนชั้นซ่อนเท่ากับ 30 โหนด ซึ่งมีค่าอัตราความสำเร็จของโครงข่ายประสาทเทียมประมาณ 0.0183 เนื่องจากเมื่อเพิ่มจำนวนโหนดเข้าไปแล้วอัตราความสำเร็จของการฝึกโครงข่ายประสาทเทียมมีค่าใกล้เคียงกัน ดังนั้นเพื่อให้การเรียนรู้ของโครงข่ายประสาทเทียมมีประสิทธิภาพดังเลือกลักษณะโครงข่ายนี้ โดยมีโครงสร้างของโครงข่ายดังแสดงในตารางที่ 4

ตารางที่ 4 แสดงลักษณะโครงสร้างของโครงข่ายประสาทเทียมในการควบคุม

โครงข่ายประสาทเทียมในการควบคุม	โครงสร้าง
จำนวนชั้น	3
จำนวนโหนดในชั้นอินพุต	3
จำนวนโหนดในชั้นซ่อน	30
จำนวนโหนดในชั้นเอาต์พุต	1
ฟังก์ชันการถ่ายโอนในชั้นซ่อน	$a = 1/(1+e^{-n})$
ฟังก์ชันการถ่ายโอนในชั้นเอาต์พุต	$a = n$
อัตราการเรียนรู้	0.01

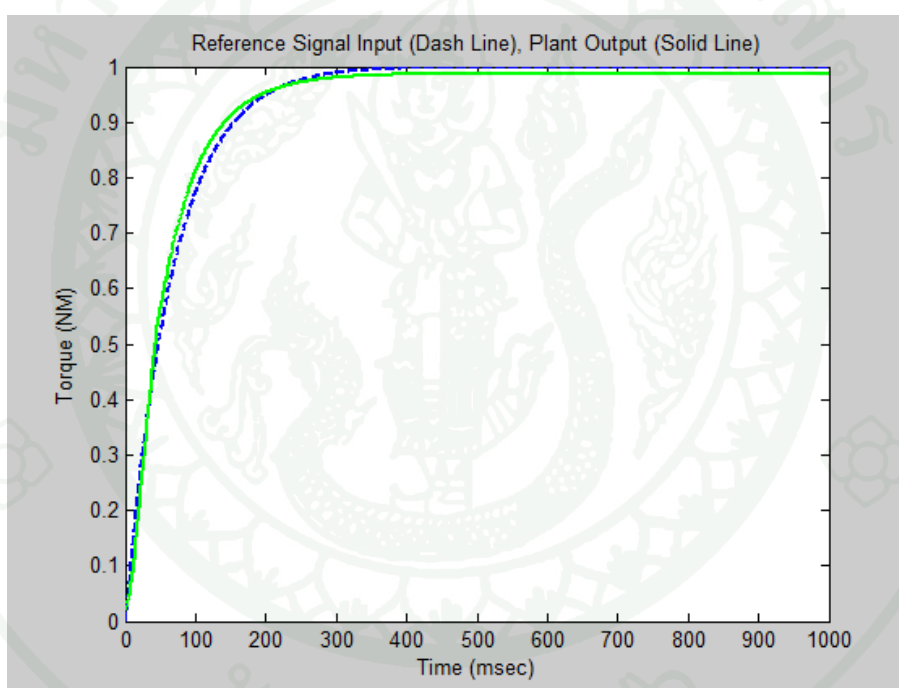
ดรรรชนีสมรรถนะขณะเรียนรู้แสดงดังภาพที่ 32 จากนั้นนำโครงสร้างโครงข่ายประสาทเทียมในการหาเอกลักษณ์และโครงข่ายประสาทเทียมในการควบคุม ไปใช้ในระบบควบคุมต่อไป



ภาพที่ 32 ดรรรชนีสมรรถนะขณะเรียนรู้ของของโครงข่ายประสาทเทียมในการควบคุมที่จำนวนโหนด เท่ากับ 30

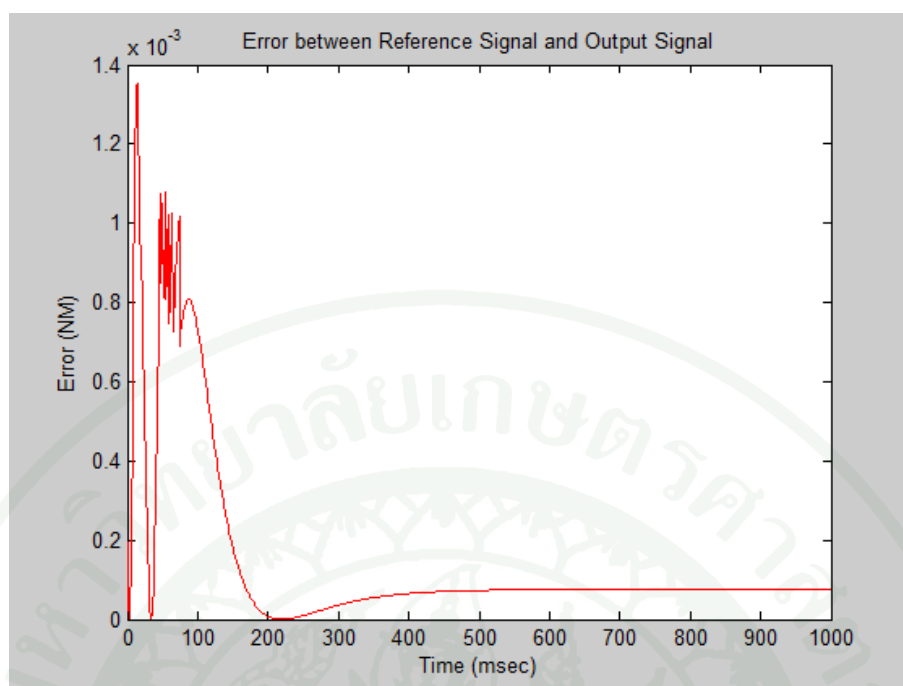
6. การจำลองการทำงานและการวิเคราะห์ผลของระบบควบคุมโดยโครงข่ายประสาทเทียม

เมื่อได้โครงข่ายประสาทเทียมในการหาเอกลักษณ์และโครงข่ายประสาทเทียมในการควบคุมจากการฝึกแล้ว จะนำโครงข่ายประสาทเทียมทั้งสองไปใช้ในระบบควบคุมเพื่อปรับตัวเอง โดยใช้ค่าถ่วงน้ำหนักและค่าไบแอสของทั้งสองโครงข่ายประสาทเทียมเป็นค่าเริ่มต้น การทำงานจะเปรียบเทียบสัญญาณเอาต์พุตที่ออกจากแพลนท์กับสัญญาณอ้างอิง ถ้าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตเกินกว่าที่กำหนดไว้ ระบบจะทำการปรับค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมในการควบคุมโดยอัตโนมัติ โดยผลการจำลองการทำงานแสดงดังภาพที่ 33

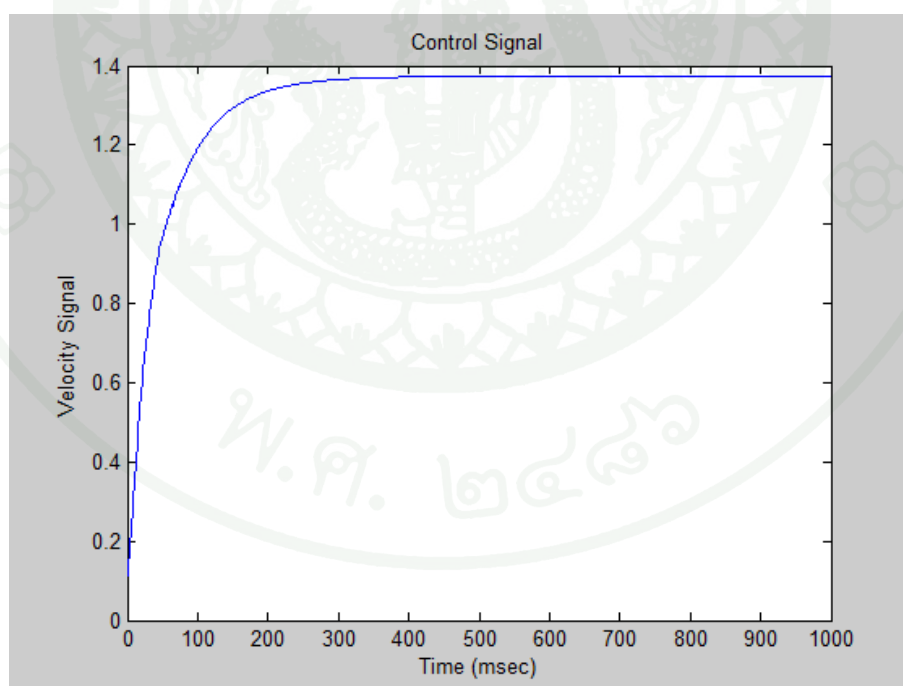


ภาพที่ 33 แสดงผลตอบสนองของระบบควบคุมที่ปรับตัวอัตโนมัติโดยใช้โครงข่ายประสาทเทียม

ความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ออกจากระบบที่กำหนดค่าไว้ 0.01 ในภาพที่ 34 ช่วงแรกของการจำลอง ค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตจะเกินค่าที่กำหนด ทำให้ระบบไปทำการปรับค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมในการควบคุมไปเรื่อยๆจนกว่าค่าความผิดพลาดจะอยู่ในขอบเขตที่กำหนด ระบบจึงหยุดการปรับค่าค่าถ่วงน้ำหนักและค่าไบแอส และใช้ค่าถ่วงน้ำหนักและค่าไบแอสนั้นใช้ควบคุมระบบแทน ภาพที่ 35 เป็นการแสดงสัญญาณควบคุมที่ส่งให้กับแพลนท์



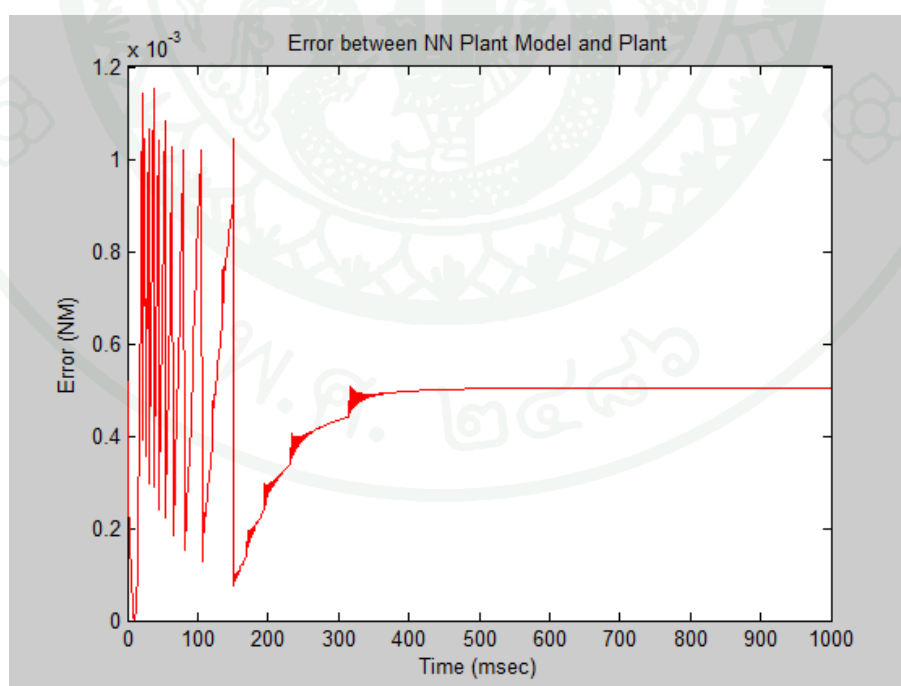
ภาพที่ 34 แสดงค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ออกจากระบบ



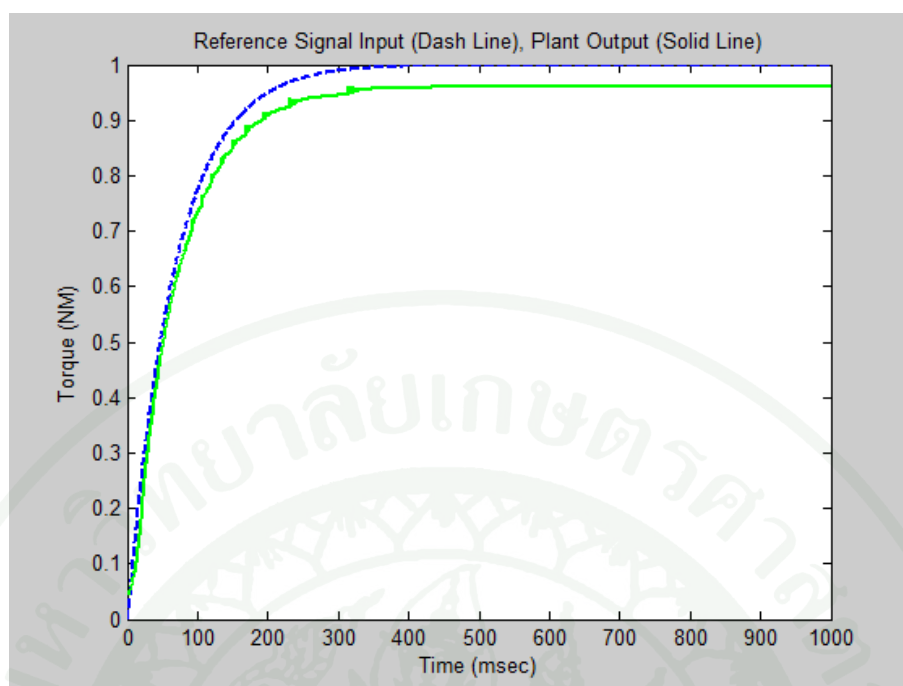
ภาพที่ 35 แสดงสัญญาณควบคุมจากโครงข่ายประสาทเทียม

ได้ทำการทดสอบสมมุติการเปลี่ยนพลานท์ของระบบโดยปรับค่าถ่วงน้ำหนักและค่าไบแอสในส่วนของพลานท์ จะทำให้เกิดความแตกต่างระหว่างโครงข่ายประสาทเทียมเอกลักษณ์และพลานท์ เมื่อมีค่าความคลาดเคลื่อนเกิดขึ้นเกินกว่าที่กำหนดไว้ ระบบจะทำการปรับค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมเอกลักษณ์ให้ใกล้เคียงกับพลานท์มากที่สุด ดังภาพที่ 36 จะเปรียบเทียบสัญญาณเอาต์พุตที่ออกจากพลานท์กับสัญญาณอ้างอิง ถ้าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตเกินกว่าที่กำหนดไว้ ระบบจะทำการปรับค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมในการควบคุมโดยอัตโนมัติ โดยผลการจำลองการทำงานแสดงดังภาพที่ 37

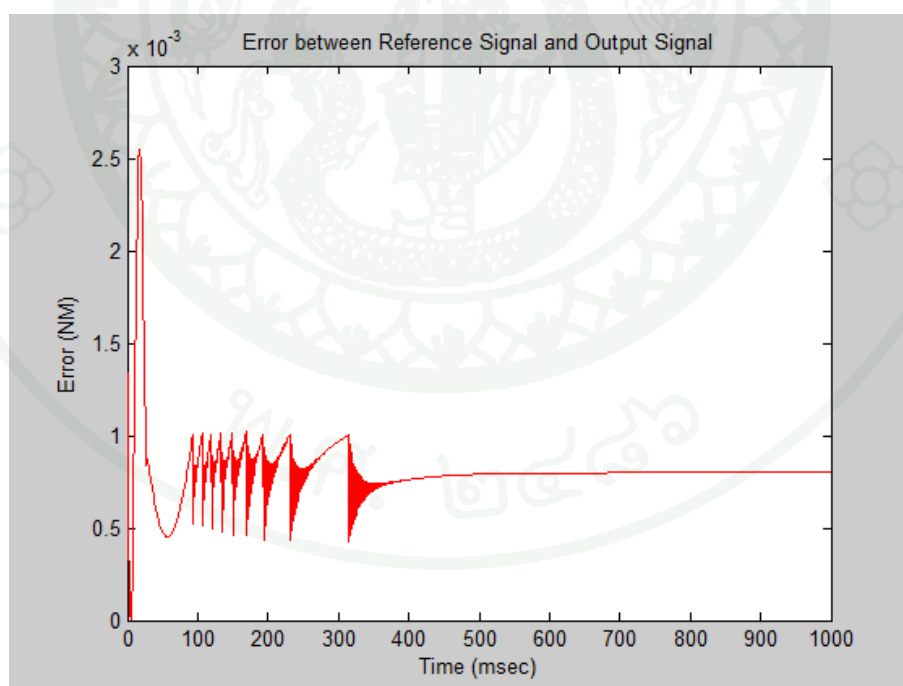
ความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ออกจากระบบที่กำหนดค่าไว้ 0.01 ในภาพที่ 38 ช่วงแรกของการจำลอง ค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตจะเกินค่าที่กำหนด ทำให้ระบบไปทำการปรับค่าถ่วงน้ำหนักและค่าไบแอสของโครงข่ายประสาทเทียมในการควบคุมไปเรื่อยๆจนกว่าค่าความผิดพลาดจะอยู่ในขอบเขตที่กำหนด ระบบจึงหยุดการปรับค่าถ่วงน้ำหนักและค่าไบแอส และใช้ค่าถ่วงน้ำหนักและค่าไบแอสนั้นใช้ควบคุมระบบแทน ภาพที่ 39 เป็นการแสดงสัญญาณควบคุมที่ส่งให้กับพลานท์



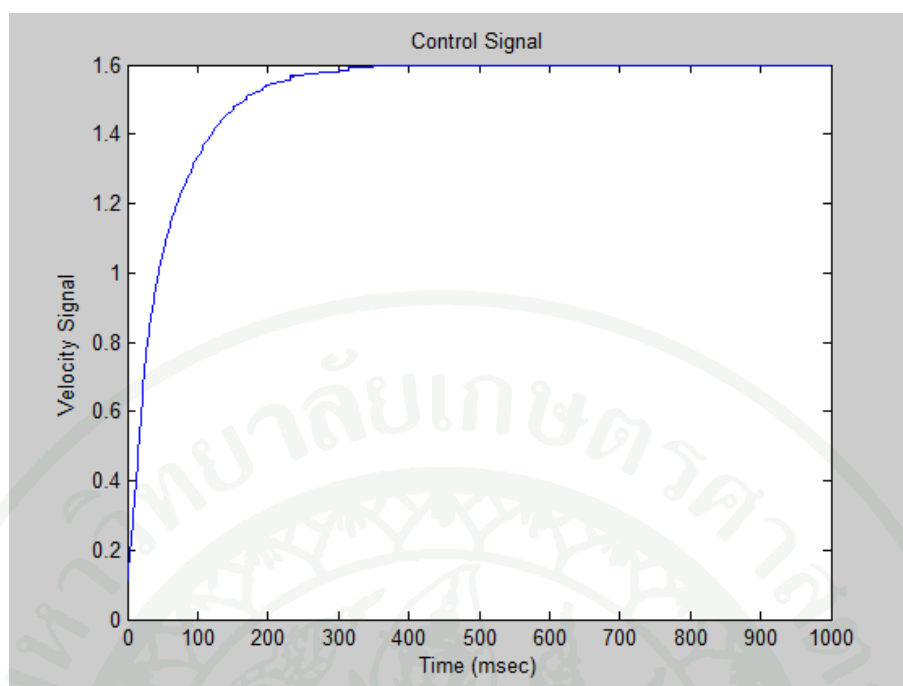
ภาพที่ 36 แสดงค่าความผิดพลาดระหว่างโครงข่ายประสาทเทียมเอกลักษณ์และพลานท์



ภาพที่ 37 แสดงผลตอบสนองของระบบควบคุมที่ปรับตัวอัตโนมัติโดยใช้โครงข่ายประสาทเทียม

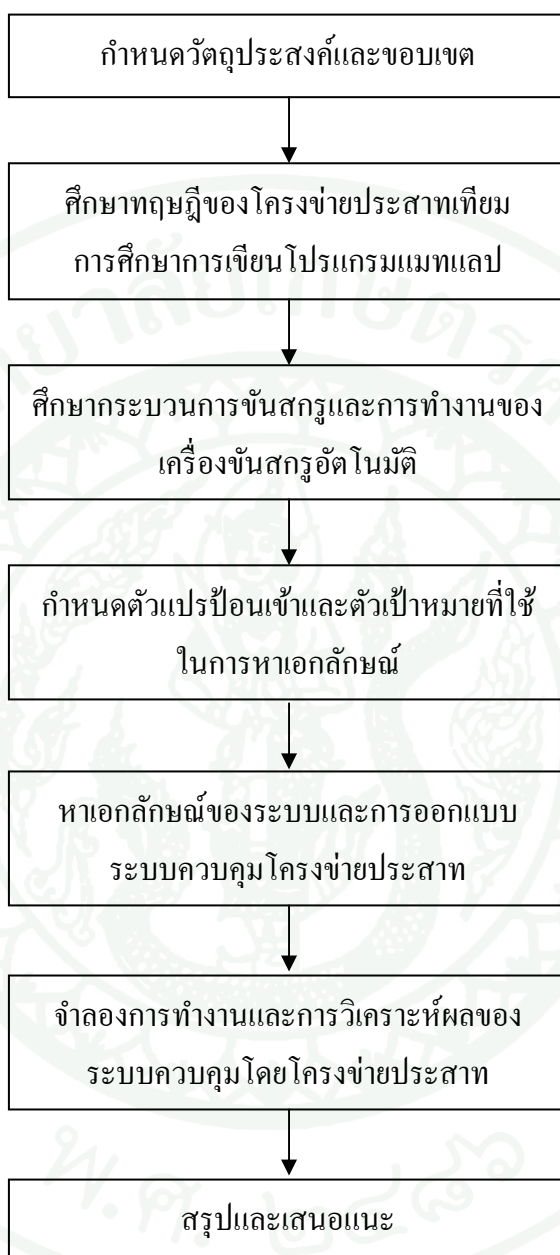


ภาพที่ 38 แสดงค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ออกจากระบบ



ภาพที่ 39 แสดงสัญญาณควบคุมจากโครงข่ายประสาทเทียม

7. สรุปขั้นตอนการทดลอง



ภาพที่ 40 แผนภูมิแสดงขั้นตอนการดำเนินงานวิจัย

ผลและวิจารณ์

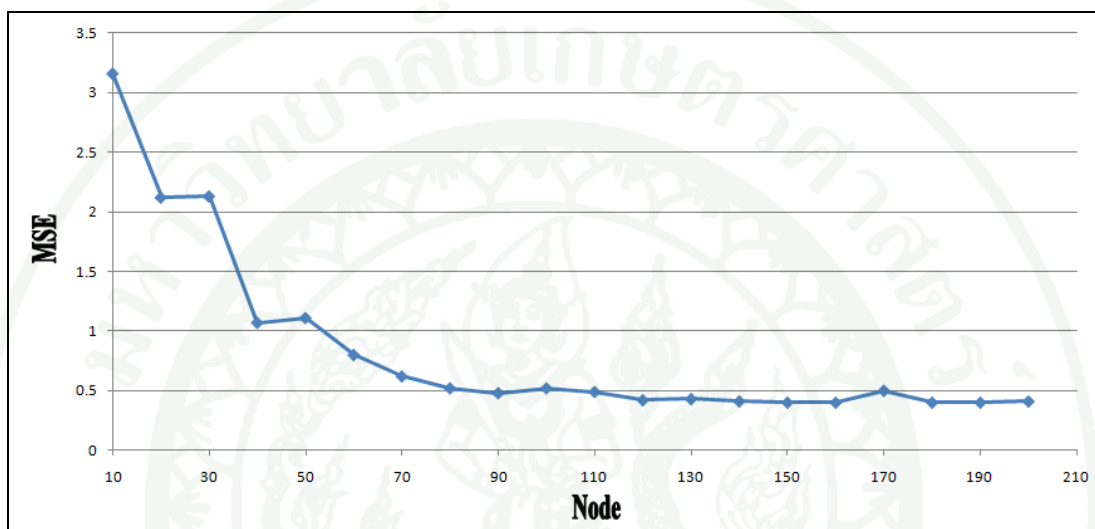
ผลการทดสอบการระบบควบคุมโดยโครงข่ายประสาทเทียม

ผลการออกแบบระบบควบคุมโดยโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ (Back-propagation Neural Network) ที่ได้จากการเขียนโปรแกรมคอมพิวเตอร์แมทแลป (Matlab) เมื่อพิจารณาถึงความสัมพันธ์ของข้อมูลป้อนเข้า (Input) ที่ใช้ในการสร้างแบบจำลองแล้ว นำข้อมูลเหล่านี้ไปใช้ในกระบวนการเรียนรู้ (Training) เพื่อใช้ในการหาเอกลักษณ์ โดยทำการแบ่งรูปแบบโครงข่ายประสาทเทียมและกำหนดโครงสร้างแต่ละรูปแบบของโครงสร้างของโครงข่ายประสาทเทียม จากนั้นทำการทดลองหาจำนวนโหนดในชั้นซ่อนที่แตกต่างกันของแต่ละรูปแบบ และต่อมาทำการเปรียบเทียบโครงสร้างของโครงข่ายประสาทเทียมในแต่ละรูปแบบในขั้นตอนของการเรียนรู้ให้มีความเหมาะสม ซึ่งพิจารณาได้จากโครงข่ายประสาทเทียมที่ให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองต่ำที่สุด (Mean Square Error : MSE) เมื่อได้โครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสมจึงทำการบันทึกค่าถ่วงน้ำหนักและค่าไบแอสที่สรุปได้จากขั้นตอนของกระบวนการเรียนรู้ จากนั้นนำค่าที่บันทึกได้เหล่านี้ไปสู่ขั้นตอนกระบวนการเรียนรู้เพื่อใช้ในการออกแบบระบบควบคุม โดยทำการแบ่งรูปแบบโครงข่ายประสาทเทียมและกำหนดโครงสร้างแต่ละรูปแบบของโครงสร้างของโครงข่ายประสาทเทียมเช่นเดียวกับกระบวนการเรียนรู้เพื่อใช้ในการหาเอกลักษณ์ จากนั้นทำการทดลองหาจำนวนโหนดในชั้นซ่อนที่แตกต่างกันของแต่ละรูปแบบและต่อมาทำการเปรียบเทียบโครงสร้างของโครงข่ายประสาทเทียมในแต่ละรูปแบบในขั้นตอนของการเรียนรู้ให้มีความเหมาะสม ซึ่งพิจารณาได้จากโครงข่ายประสาทเทียมที่ให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองต่ำที่สุด เมื่อได้โครงข่ายประสาทเทียมในการควบคุมที่เหมาะสมจึงทำการบันทึกค่าถ่วงน้ำหนักและค่าไบแอส แล้วนำค่าถ่วงน้ำหนักและค่าไบแอสของทั้งสองโครงข่ายไปใช้ในการควบคุมระบบต่อไป

กระบวนการเรียนรู้ของโครงข่ายประสาทเทียมใช้การหาเอกลักษณ์ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของแต่ละโครงสร้าง แสดงดังตารางที่ 1 โดยกราฟแสดงค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ของแต่ละรูปแบบแสดงดังภาพที่ 37

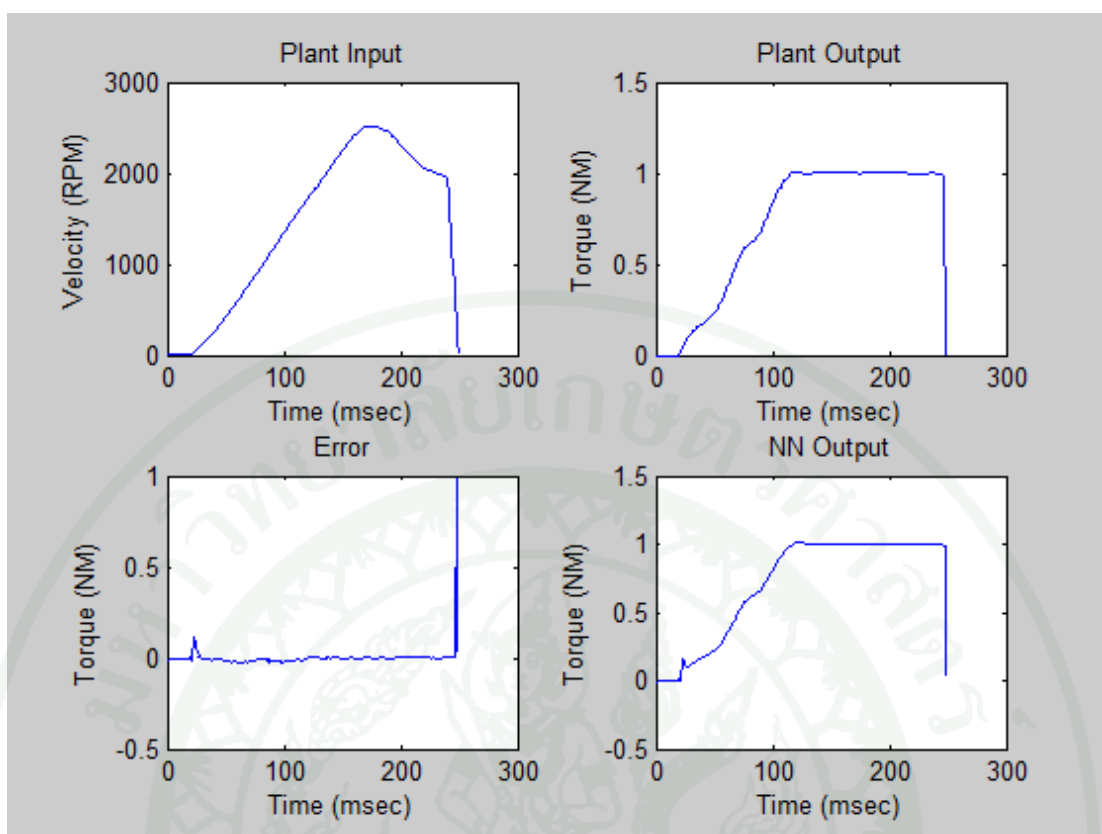
จากภาพที่ 37 พิจารณารูปแบบโครงสร้างของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ทั้ง 20 รูปแบบ เมื่อเพิ่มจำนวนโหนดในชั้นซ่อนมากขึ้นเรื่อยๆตั้งแต่ 10 – 200 โหนด โดยเพิ่มจำนวนโหนดครั้งละ 10 โหนด จะส่งผลทำให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของ

กระบวนการเรียนรู้ในการหาเอกลักษณ์มีแนวโน้มลดลงเรื่อยๆ จนกระทั่งถึงโครงสร้างที่มีจำนวน โหนดในชั้นซ่อนเป็น 150 โหนด และโครงสร้างหลังจากนี้จะให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองมีค่าใกล้เคียงกัน และมีแนวโน้มคงที่ จนถึงโครงสร้างที่มีจำนวน โหนดเท่ากับ 200 โหนด ซึ่ง กราฟแสดงให้เห็นว่าหลังจากเพิ่มจำนวนโหนดมากกว่า 150 โหนดแล้ว พบว่าไม่สามารถทำให้ โครงข่ายประสาทเทียมเรียนรู้ได้ดีขึ้นมากกว่าเดิม



ภาพที่ 41 ค่าบรรจบกันของผลการเรียนรู้ของโครงข่ายประสาทเทียมในการหาเอกลักษณ์

เมื่อได้โครงสร้างของโครงข่ายประสาทเทียมที่เหมาะสมคือรูปแบบที่มีโครงสร้าง $2 \times 150 \times 1$ จากนั้นจึงทำการบันทึกค่าถ่วงน้ำหนัก และค่าไบแอสที่ได้จากขั้นตอนของกระบวนการเรียนรู้ ซึ่งมีผลการทดสอบหลังจากได้รับการฝึกดังภาพที่ 38 จากนั้นนำโครงสร้างโครงข่ายประสาทเทียมเอกลักษณ์นี้ไปใช้ในการเรียนรู้โครงข่ายประสาทเทียมในการควบคุมต่อไป แสดงค่าถ่วงน้ำหนักและค่าไบแอสที่เหมาะสมดังตารางที่ 5 -8



ภาพที่ 42 ผลการทดสอบโครงข่ายประสาทเทียมในการหาเอกลักษณ์

$IW_{j,i}$ = ค่าน้ำหนักการเชื่อมต่อระหว่างอินพุต i ในชั้นรับข้อมูลไปยังโหนด j ในชั้นซ่อน

ตารางที่ 5 ค่าน้ำหนักของชั้นรับข้อมูลไปยังชั้นซ่อนของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสม

Neuron	Input	
	$i = 1$	$i = 2$
$j = 1$	-2.1666	0.4593
2	0.4221	-0.8181
3	-0.2948	-1.3625
4	-0.3425	-0.1232
5	0.6477	1.6012
6	-0.3035	-0.3893
7	-0.2268	0.4085

ตารางที่ 5 (ต่อ)

Neuron \ Input	i = 1	i = 2
8	-0.8716	-0.5989
9	1.1258	-2.0485
10	0.9916	0.1538
11	0.6864	1.6938
12	0.5317	0.1397
13	0.7264	-1.1512
14	0.6943	-0.9903
15	0.3941	0.3582
16	0.6316	-0.3465
17	-0.7091	0.9860
18	0.0293	-0.2891
19	0.5504	-2.0860
20	-0.7255	-0.1738
21	0.0008	0.7115
22	-0.1875	1.0262
23	1.2414	1.2850
24	0.4725	0.9995
25	1.0945	1.0755
26	0.6227	1.8961
27	1.0092	1.5375
28	-0.0945	0.2455
29	0.1152	-0.4776
30	-0.6282	-0.3298
31	0.3876	0.7016
32	3.0743	1.2080
33	0.0981	0.1543
34	-0.0456	1.5127
35	0.8583	-0.7221

ตารางที่ 5 (ต่อ)

Neuron \ Input	i = 1	i = 2
36	-1.5636	-0.4547
37	1.1287	0.1736
38	-0.8476	-0.6621
39	0.5675	0.8717
40	0.2770	0.1913
41	-0.0670	1.8498
42	-0.7172	-0.2838
43	0.4173	1.8663
44	1.1246	-2.4296
45	0.6822	-0.1252
46	1.3961	0.4570
47	-1.3886	1.9377
48	0.0139	0.3433
49	-1.3504	1.7741
50	0.1146	1.8931
51	0.5416	-0.6199
52	-0.7677	-1.1928
53	1.2828	0.8583
54	-0.4749	-0.8149
55	0.6241	0.4046
56	-0.3477	-0.5793
57	-0.7592	-0.3768
58	-0.7372	0.5963
59	-1.4510	0.7122
60	1.5207	0.3144
61	-2.4472	-0.9442
62	-1.2900	-0.8471
63	-1.9846	0.1849

ตารางที่ 5 (ต่อ)

Neuron \ Input	i = 1	i = 2
64	-1.3429	-2.1815
65	-2.7918	-1.1752
66	0.6022	-1.5663
67	0.0404	-0.3297
68	-1.0309	0.4914
69	-1.3010	2.3615
70	-1.3180	-0.1564
71	-0.2785	-1.0710
72	-2.0221	-0.4305
73	0.4067	0.5195
74	-0.0898	1.0250
75	-0.3423	-0.4595
76	0.6522	-0.1659
77	0.3457	1.1855
78	-0.3524	-0.2952
79	0.0307	0.1944
80	-0.1333	-0.4188
81	-0.6260	-0.4400
82	1.1114	1.0574
83	1.2022	-0.5409
84	0.0727	0.0457
85	-1.0267	-1.0289
86	-1.1730	0.6054
87	0.2510	0.4310
88	-0.2646	0.4403
89	-0.7981	0.0002
90	0.1320	-0.7433
91	-1.1846	-0.1309

ตารางที่ 5 (ต่อ)

Neuron \ Input	i = 1	i = 2
92	0.3404	-1.2858
93	-1.5921	-0.4287
94	-2.4538	2.3078
95	-0.2081	1.6753
96	-0.9819	0.8670
97	1.5311	0.4293
98	1.3727	0.8256
99	-0.6121	0.3742
100	-0.9667	0.9343
101	-0.2164	1.2040
102	-0.3161	-1.0211
103	1.3853	-0.2655
104	-0.3482	0.2771
105	1.2055	-0.5446
106	1.2372	-0.9920
107	2.5197	1.2719
108	-0.3795	0.6664
109	-0.4753	-0.9131
110	1.6757	-1.0519
111	-0.7384	0.4011
112	-1.8155	-0.6894
113	0.0867	-1.1978
114	0.5003	-0.3536
115	-0.3630	-0.8464
116	-0.3507	0.8491
117	-1.3629	2.3053
118	1.7228	-1.6987
119	0.2014	0.8359

ตารางที่ 5 (ต่อ)

Neuron \ Input	i = 1	i = 2
120	-0.5521	-0.2899
121	0.4018	-0.8411
122	0.8344	0.9615
123	-1.0483	-0.8862
124	0.2253	-1.8815
125	-0.4824	-1.1242
126	1.6507	0.5782
127	0.1589	0.4713
128	-1.1247	1.0058
129	-0.1118	-0.8891
130	0.4491	1.2603
131	2.1006	-2.0040
132	0.0396	-0.4195
133	0.5716	-0.3589
134	1.2008	-0.1812
135	0.5088	1.0687
136	-0.8698	-0.6768
137	-0.1611	0.4167
138	-0.6578	-1.9116
139	0.7068	-0.9199
140	1.5403	0.9391
141	-0.0219	-1.7444
142	1.0548	1.4146
143	-0.6968	-0.4868
144	0.1593	0.4805
145	-0.1765	-1.1767
146	1.6303	0.7548
147	0.2940	-0.3238

ตารางที่ 5 (ต่อ)

Neuron	Input	
	i = 1	i = 2
148	0.7677	1.2571
149	-0.3412	-0.5473
150	-0.6334	-0.7338

LW_{kj} = คำนวณน้ำหนักการเชื่อมต่อระหว่างโหนด j ในชั้นรับข้อมูลไปยังโหนด k ในชั้นเอาต์พุต

ตารางที่ 6 น้ำหนักของชั้นซ่อนไปยังชั้นเอาต์พุตของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสม

Neuron	Output	
	j = 1	k = 1
1	-0.6669	
2	1.7969	
3	-0.4058	
4	-1.4772	
5	0.9703	
6	-0.0846	
7	-0.1894	
8	0.3555	
9	0.7411	
10	-0.6336	
11	0.1445	
12	-0.4600	
13	-0.6320	
14	0.2896	
15	0.6789	
16	-1.4638	

ตารางที่ 6 (ต่อ)

Neuron	Output	k =1
17		1.1940
18		-0.0594
19		-0.2920
20		-1.7187
21		-0.2036
22		-0.0643
23		0.2308
24		-0.3962
25		0.9609
26		1.6916
27		1.3024
28		-0.1923
29		0.2878
30		0.3037
31		-0.5863
32		0.0636
33		-0.5974
34		0.0778
35		1.5333
36		-0.3160
37		-0.8246
38		1.2573
39		0.8920
40		-0.5481
41		-0.7266
42		1.3463
43		0.1154

ตารางที่ 6 (ต่อ)

Neuron	Output	k =1
44		0.0914
45		0.5052
46		-0.4965
47		1.1210
48		2.3275
49		-2.4877
50		0.2284
51		0.8770
52		-0.9059
53		0.1734
54		0.0185
55		-0.5791
56		0.5547
57		0.9710
58		0.2305
59		0.4469
60		-0.4836
61		1.0129
62		-0.1302
63		0.4823
64		1.0407
65		-0.3690
66		-0.3628
67		0.0568
68		-0.1535
69		-1.1991
70		0.3434

ตารางที่ 6 (ต่อ)

Neuron	Output	k =1
71		-1.5182
72		-0.4427
73		-1.3164
74		1.3470
75		-0.0016
76		0.2769
77		-0.2965
78		1.3344
79		-0.3627
80		1.0370
81		1.0410
82		-0.4005
83		0.2693
84		0.0411
85		2.0310
86		-0.9508
87		-1.5262
88		0.2795
89		-0.9198
90		-0.6692
91		0.0788
92		-1.4846
93		0.9445
94		0.4170
95		1.4341
96		-0.3022
97		0.8995

ตารางที่ 6 (ต่อ)

Neuron	Output	k =1
98		-1.1607
99		0.2846
100		1.1874
101		-0.6111
102		-0.0058
103		1.6114
104		-0.6383
105		-1.2299
106		0.8448
107		0.0361
108		-1.1140
109		-1.0084
110		0.2388
111		0.3330
112		-0.4315
113		0.1249
114		0.4329
115		-0.8613
116		-0.3458
117		-0.7406
118		1.2882
119		-1.3085
120		1.4473
121		-0.7690
122		0.0347
123		0.8728
124		0.7110

ตารางที่ 6 (ต่อ)

Neuron	Output	k =1
125		-0.4304
126		0.8657
127		-1.1094
128		1.4541
129		1.2494
130		0.1221
131		0.3230
132		0.8439
133		-0.3533
134		0.2489
135		0.7607
136		0.2229
137		-1.3675
138		-0.4243
139		-2.1138
140		-0.3628
141		-0.9806
142		0.0124
143		0.3051
144		0.9390
145		-0.2233
146		0.1134
147		0.8934
148		1.3707
149		0.1360
150		0.7662

b_j^1 = ค่าไบแอสของโหนด j ในชั้นซ่อน

ตารางที่ 7 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมในการหาเอกลักษณ์ที่เหมาะสม

Neuron	ค่าไบแอส
$j=1$	-0.6904
2	1.5289
3	-0.6449
4	-0.7790
5	0.4903
6	-0.0638
7	-1.5626
8	1.0592
9	0.6573
10	1.6856
11	0.2892
12	0.7540
13	-1.1271
14	1.8500
15	2.3594
16	-0.3019
17	-0.7471
18	0.9965
19	-1.1244
20	0.2601
21	-0.1214
22	0.8307
23	-1.9720
24	0.1468
25	0.2131
26	-1.8105

ตารางที่ 7 (ต่อ)

Neuron	ค่าไบแอส
27	-0.2247
28	0.6469
29	-1.2875
30	-0.3229
31	0.2300
32	-0.5228
33	1.6206
34	0.7670
35	-0.6562
36	0.3452
37	1.2046
38	0.1784
39	0.5136
40	0.7074
41	0.7753
42	-1.3968
43	-0.6792
44	-0.3604
45	-1.6451
46	-1.3685
47	0.3004
48	0.2705
49	1.0352
50	0.3555
51	-0.8093
52	0.0295
53	-0.7216
54	0.8627

ตารางที่ 7 (ต่อ)

Neuron	ค่าไบแอส
55	-1.4751
56	0.8304
57	0.3959
58	0.7634
59	-0.5974
60	0.2411
61	-0.0982
62	-1.5073
63	0.4539
64	0.3868
65	-0.7274
66	1.2497
67	-1.7673
68	-0.3997
69	-0.3271
70	-0.7547
71	-0.5913
72	0.5032
73	-0.8072
74	-0.0007
75	-0.4275
76	-0.1722
77	1.9397
78	-0.6021
79	0.9194
80	-0.0279
81	-0.0319
82	-1.8913

ตารางที่ 7 (ต่อ)

Neuron	ค่าไบแอส
83	1.0053
84	0.1524
85	-0.5399
86	0.0908
87	-0.0104
88	1.8248
89	-1.0424
90	-0.5302
91	0.2366
92	1.8868
93	0.1968
94	1.0944
95	0.6846
96	0.7346
97	1.2540
98	2.3124
99	0.1336
100	-0.3486
101	0.2218
102	-0.3055
103	-2.3463
104	-0.0371
105	0.2303
106	-0.4127
107	-0.2161
108	1.2786
109	0.1328
110	-0.0458

ตารางที่ 7 (ต่อ)

Neuron	ค่าไบแอส
111	-0.6760
112	-0.5378
113	-0.1106
114	0.0925
115	0.4288
116	1.6458
117	1.3843
118	-1.2488
119	1.7924
120	-1.0025
121	2.5803
122	0.1996
123	-0.4857
124	-2.0383
125	-0.3781
126	1.5148
127	-0.3479
128	-0.5204
129	-1.3773
130	-1.1757
131	-0.5972
132	0.3034
133	-2.0546
134	0.6191
135	-0.5159
136	0.7905
137	0.0844
138	-0.4231

ตารางที่ 7 (ต่อ)

Neuron	ค่าไบแอส
139	0.6027
140	0.5251
141	2.0431
142	-1.2990
143	-0.7187
144	-1.2587
145	-0.5611
146	-0.5081
147	-0.5190
148	-0.5339
149	-0.5195
150	-0.2803

b_k^2 = ค่าไบแอสของโหนด k ในชั้นเอาต์พุต

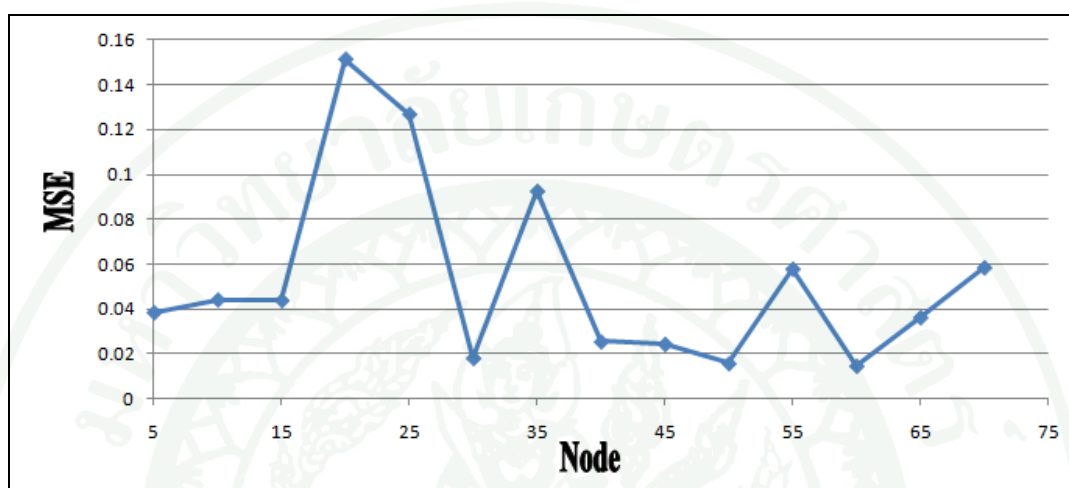
ตารางที่ 8 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมที่เหมาะสม

Neuron	ค่าไบแอส
$k=1$	0.3663

กระบวนการเรียนรู้ของโครงข่ายประสาทเทียมในการหาแบบระบบควบคุมค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของแต่ละ โครงสร้าง แสดงดังตารางที่ 3 โดยกราฟแสดงค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ของแต่ละรูปแบบแสดงดังภาพที่ 39

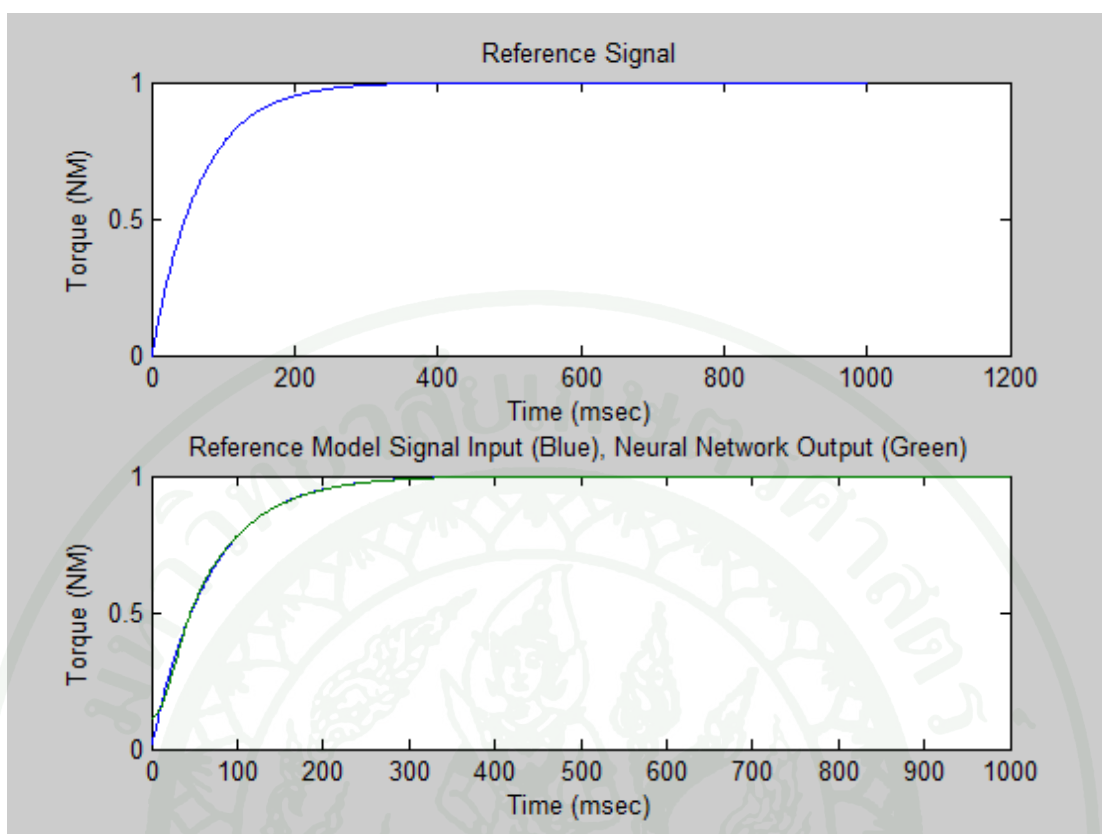
จากภาพที่ 39 พิจารณารูปแบบโครงสร้างของโครงข่ายประสาทเทียมในการหาแบบควบคุมทั้ง 14 รูปแบบ เมื่อเพิ่มจำนวนโหนดในชั้นซ่อนมากขึ้นเรื่อยๆตั้งแต่ 5 – 70 โหนด โดยเพิ่มจำนวนโหนดครั้งละ 5 โหนด จะส่งผลทำให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการ

เรียนรู้ในการหาระบบควบคุมมีแนวโน้มลดลงเรื่อยๆ จนกระทั่งถึงโครงสร้างที่มีจำนวนโหนดในชั้นซ่อนเป็น 30 โหนด และโครงสร้างหลังจากนี้จะให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองมีแนวโน้มเพิ่มขึ้นเรื่อยๆ และคงที่ ซึ่งกราฟแสดงให้เห็นว่าหลังจากเพิ่มจำนวนโหนดมากกว่า 30 โหนดแล้ว พบว่าไม่สามารถทำให้โครงข่ายประสาทเทียมเรียนรู้ได้ดีขึ้นมากกว่าเดิม



ภาพที่ 43 ค่าครุชนิสมรรถนะของผลการเรียนรู้ของโครงข่ายประสาทเทียมในการหาระบบควบคุม

เมื่อได้โครงสร้างของโครงข่ายประสาทเทียมในการหาระบบควบคุมที่เหมาะสมคือรูปแบบที่มีโครงสร้าง $2 \times 30 \times 1$ จากนั้นจึงทำการบันทึกค่าถ่วงน้ำหนักและค่าไบแอสที่ได้จากขั้นตอนของกระบวนการเรียนรู้ ซึ่งมีผลการทดสอบหลังจากได้รับการฝึกดังภาพที่ 40 จากนั้นจากนั้นนำโครงสร้างโครงข่ายประสาทเทียมในการหาเอกลักษณ์และโครงข่ายประสาทเทียมในการควบคุม ไปใช้ในระบบควบคุมต่อไป แสดงค่าถ่วงน้ำหนักและค่าไบแอสที่เหมาะสมดังตารางที่ 9 – 12



ภาพที่ 44 ผลการทดสอบโครงข่ายประสาทเทียมในการควบคุม

$IW_{j,i}$ = ค่าน้ำหนักการเชื่อมต่อระหว่างอินพุต i ในชั้นรับข้อมูลไปยังโหนด j ในชั้นซ่อน

ตารางที่ 9 ค่าน้ำหนักของชั้นรับข้อมูลไปยังชั้นซ่อนของโครงข่ายประสาทเทียมในการหาระบบควบคุมที่เหมาะสม

Neuron	Input		
		$i = 1$	$i = 2$
$j = 1$		0.5573	0.4633
2		1.9496	-0.8587
3		1.6694	-0.3834
4		1.0482	-0.5375
5		-0.6722	-0.2348
6		-0.8091	-0.3824
7		0.1695	-0.8474

ตารางที่ 9 (ต่อ)

Neuron	Input	
	i = 1	i = 2
8	-0.4042	-0.9042
9	0.8994	0.5334
10	1.3251	1.6826
11	-0.2412	-1.5317
12	-0.3136	-0.5162
13	0.8843	0.1274
14	0.5712	0.5685
15	-1.8498	0.7816
16	-0.6609	0.2837
17	-0.7684	-1.7585
18	-1.3231	0.6706
19	-0.2071	0.8722
20	0.2259	-0.4537
21	1.3953	-0.3159
22	0.5959	-0.6942
23	1.1535	-0.1737
24	-0.7520	-0.9335
25	0.3274	1.1939
26	1.7050	-1.1424
27	-0.5747	-0.3075
28	0.3771	1.0227
29	1.1728	0.2840
30	-1.0526	-0.3748

LW_{kj} = คำนวณน้ำหนักการเชื่อมต่อระหว่างโหนด j ในชั้นรับข้อมูลไปยังโหนด k ในชั้นเอาต์พุต

ตารางที่ 10 น้ำหนักของชั้นซ่อนไปยังชั้นเอาต์พุตของโครงข่ายประสาทเทียมในการหาระบบ
ควบคุมที่เหมาะสม

Neuron	Output	k =1
j =1		-0.2000
2		0.6066
3		1.1400
4		0.8391
5		1.7213
6		-1.0770
7		-0.2033
8		-2.0924
9		0.6455
10		1.0222
11		-0.0811
12		-1.6617
13		-1.8114
14		2.0184
15		-1.2386
16		0.9374
17		-0.4131
18		-0.0465
19		-0.4301
20		0.5920
21		0.0489
22		-0.9512
23		-0.1035
24		-0.1386
25		0.1538
26		0.6082
27		0.3872

ตารางที่ 10 (ต่อ)

Neuron	Output	k=1
28		-2.2035
29		-0.5401
30		-0.2242

b_j^1 = ค่าไบแอสของโหนด j ในชั้นซ่อน

ตารางที่ 11 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมในการหาระบบควบคุมที่เหมาะสม

Neuron	ค่าไบแอส
j=1	0.4716
2	0.6647
3	0.4993
4	-0.3500
5	1.6052
6	-0.7753
7	-1.4687
8	0.3823
9	0.9296
10	0.8743
11	0.5288
12	0.7918
13	0.1681
14	1.5024
15	1.2733
16	0.1271
17	0.7966
18	1.1101

ตารางที่ 11 (ต่อ)

Neuron	ค่าไบแอส
19	1.5025
20	-0.1127
21	1.2522
22	1.0634
23	0.9732
24	1.2047
25	-0.1179
26	0.6207
27	1.1251
28	-0.4898
29	-1.3418
30	0.6589

b_k^2 = ค่าไบแอสของโหนด k ในชั้นเอาต์พุต

ตารางที่ 12 ค่าไบแอสของชั้นซ่อนของโครงข่ายประสาทเทียมที่เหมาะสม

Neuron	ค่าไบแอส
k =1	0.3569

เมื่อได้โครงข่ายประสาทเทียมในการหาเอกลักษณ์และโครงข่ายประสาทเทียมในการควบคุมจากการฝึกแล้ว จะนำโครงข่ายประสาทเทียมทั้งสองไปใช้ในระบบควบคุมเพื่อปรับตัวเอง โดยใช้ค่าถ่วงน้ำหนักและค่าไบแอสของทั้งสองโครงข่ายประสาทเทียมเป็นค่าเริ่มต้น โดยผลการจำลองการทำงานแสดงดังภาพที่ 33 โดยค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาต์พุตที่ได้จากการทดสอบมีค่าเท่ากับ -0.009 นิวตันเมตร ดังภาพที่ 34 ซึ่งขอบเขตของที่กำหนดอยู่ที่ ± 0.07 นิวตันเมตร ดังนั้นผลการควบคุมของระบบนี้จึงอยู่ในขอบเขตข้อกำหนด

สรุปและข้อเสนอแนะ

สรุป

การควบคุมตามแบบจำลองอ้างอิงโดยการใช้โครงข่ายประสาทเทียมเป็นรูปแบบการประยุกต์การใช้งานโครงข่ายประสาทเทียมในระบบควบคุมอีกวิธีหนึ่ง โดยลักษณะตัวควบคุมดังกล่าวไม่จำเป็นต้องเขียนให้อยู่ในรูปของสมการทางคณิตศาสตร์ที่ชัดเจน แต่อาศัยวิธีการเรียนรู้และชุดข้อมูลที่เหมาะสมในการฝึกโครงข่ายประสาทเทียมเพื่อใช้เป็นตัวควบคุม ความสามารถของโครงข่ายประสาทในการเรียนรู้ จัดจำลักษณะความสัมพันธ์ของข้อมูล และสามารถให้ผลตอบที่สอดคล้องกับชุดของข้อมูลที่ป้อนเข้าหลังจากที่ได้รับการฝึกฝนอย่างพอเพียง จึงสามารถนำไปประยุกต์ใช้กับการควบคุมแบบปรับตัวเองได้ เนื่องจากไม่ทราบสัญญาณควบคุมที่เหมาะสมในการป้อนเข้าสู่ระบบเพื่อให้ระบบเป็นไปตามต้องการ ดังนั้นการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมจึงอาศัยวิธีส่งผ่านค่าความคลาดเคลื่อนระหว่างสัญญาณอ้างอิงและสัญญาณที่ออกจากระบบ ผ่านกลับไปตามชั้นต่างๆของโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ ค่าความคลาดเคลื่อนที่แผ่กลับมามีใช้ในการฝึกโครงข่ายประสาทเพื่อเป็นตัวควบคุมให้สามารถเรียนรู้และลดค่าความคลาดเคลื่อนดังกล่าว ซึ่งงานวิจัยนี้ได้นำเสนอการประยุกต์ใช้หลักการและทฤษฎีของโครงข่ายประสาทเทียมในการควบคุมการขึ้นสกรู เพื่อหาระบบควบคุมให้กับเครื่องขึ้นสกรูอัตโนมัติ โดยประกอบด้วยโครงข่ายประสาทเทียม 2 ส่วน คือ ส่วนของการหาเอกลักษณ์ และส่วนเป็นตัวควบคุม ซึ่งใช้หลักการของโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ และมีการเรียนรู้แบบชี้แนะเพื่อทำการปรับเปลี่ยนค่าน้ำหนักและค่าไบแอสของโครงข่ายเพื่อให้ผลลัพธ์ของแบบจำลองใกล้เคียงกับค่าความจริงมากที่สุด

โดยทำการแบ่งการเรียนรู้ของโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ ออกเป็น 20 รูปแบบ และการแบ่งการเรียนรู้ของโครงข่ายประสาทที่ใช้เป็นตัวควบคุม ออกเป็น 14 รูปแบบ แล้วทดลองหาจำนวนโหนดในชั้นซ่อนในรูปแบบที่แตกต่างกับเพื่อออกแบบโครงข่ายประสาทเทียมที่ทำให้ค่าความคลาดเคลื่อนต่ำสุด ซึ่งการดำเนินงานสรุปได้ดังนี้

1. โครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ มีปัจจัยป้อนเข้าพลาตคือ ค่าความเร็วของมอเตอร์แกน R ส่วนชุดเป้าหมายเป็นค่าที่ออกจากพลาตคือ ค่าทอร์กของการขึ้นสกรู โดยโครงสร้างที่เหมาะสมที่ทำให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ต่ำสุดคือโครงสร้างที่มีชั้นซ่อน 150 โหนด ฟังก์ชันการถ่ายโอนในชั้นซ่อนและฟังก์ชันการถ่ายโอนใน

ชั้นเอาท์พุทเป็นลือก-ซิกมอยด์และลิเนียร์ตามลำดับ โดยค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ คือ 0.3963 นิวตันเมตร และมีผลการทดสอบหลังจากได้รับการฝึกดังภาพที่ 39

2. โครงข่ายประสาทที่ใช้เป็นตัวควบคุมไม่สามารถกำหนดเป้าหมายได้โดยตรง ต้องอาศัยการส่งค่าผ่านตามชั้นต่างๆของโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ โดยโครงสร้างที่เหมาะสมที่ทำให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ต่ำสุด คือโครงสร้างที่มีชั้นซ่อน 30 โหนด ฟังก์ชันการถ่ายโอนในชั้นซ่อนและฟังก์ชันการถ่ายโอนในชั้นเอาท์พุทเช่นเดียวกับโครงข่ายประสาทที่ใช้ในการหาเอกลักษณ์ คือลือก-ซิกมอยด์และลิเนียร์ตามลำดับ โดยค่าเฉลี่ยความคลาดเคลื่อนกำลังสองของกระบวนการเรียนรู้ คือ 0.0183 นิวตันเมตร และมีผลการทดสอบหลังจากได้รับการฝึกดังภาพที่ 40

3. การทดสอบการควบคุมโดยนำโครงข่ายประสาทเทียมทั้งสองที่ได้รับฝึกแล้ว มาเป็นค่าเริ่มต้นของระบบ โดยค่าน้ำหนักและค่าของไบแอสของโครงข่ายประสาทที่ใช้เป็นตัวควบคุมจะเปลี่ยนไปเมื่อระบบมีการเปลี่ยนแปลง โดยค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับเอาท์พุทที่ได้จากการทดสอบมีค่าเท่ากับ -0.009 ซึ่งอยู่ในขอบเขตที่กำหนดคือ ± 0.07 ดังแสดงในภาพที่ 34

ข้อเสนอแนะ

1. ไม่มีกฎเกณฑ์แน่นอนในการกำหนดโครงสร้างของโครงข่ายประสาทเทียมอันได้แก่จำนวนชั้น หรือจำนวนปม ที่เหมาะสมในการนำไปประยุกต์ใช้กับปัญหานั้นๆ ทั้งนี้อาศัยการลองผิดลองและนำผลที่ได้เปรียบเทียบกับกัน เพื่อให้ได้โครงสร้างที่เหมาะสมที่สุด

2. ใช้เวลาในการประมวลผลนาน ทั้งนี้ขึ้นอยู่กับความซับซ้อนของเครือข่ายโดยต้องคำนึงถึงเป็นอย่างมากในการประยุกต์ในการทำงานในเวลาจริง

3. ข้อมูลที่เข้ามาด้านนอกเหนือจากที่เคยได้เรียนรู้ในขั้นตอนการฝึกเครือข่าย อาจทำให้การควบคุมไม่เป็นไปตามความต้องการได้ ทั้งนี้โครงข่ายประสาทเทียมจะต้องใช้เวลาเรียนรู้ชุดข้อมูลนี้พอสมควร

เอกสารและสิ่งอ้างอิง

- กิดชอบ ไวยสุศรี. 2544. ระบบวัดแรงบิดและความเร็วแบบปราศจากเซนเซอร์ สำหรับการทดสอบเครื่องจักรกลไฟฟ้า. วิทยานิพนธ์ปริญญาโท, จุฬาลงกรณ์มหาวิทยาลัย.
- ชวพงษ์ สิงห์แพทย์. 2531. การควบคุมอัตโนมัติ. คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์, กรุงเทพฯ.
- ณัฐภ นิมปีดิวัน. 2550. **Neural Network and Artificial Intelligent**. คณะวิศวกรรมศาสตร์ มหาวิทยาลัยกรุงเทพ, กรุงเทพฯ.
- นวกัก เอื้ออนันต์. 2546. เอกสารประกอบการเรียนวิชาวงจรขั้วนิรอลเทียม. คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น, ขอนแก่น.
- รดาณูช วิเศษสรโรโชค. 2544. การสร้างแบบจำลองและการออกแบบตัวควบคุมแบบขั้วงานนิรอลของตัวย่อยเยื่อกระดาษ. วิทยานิพนธ์ปริญญาโท, จุฬาลงกรณ์มหาวิทยาลัย.
- วิศณีย์ ตั้งยืนยง. 2544. การสร้างแบบจำลองฟัซซี่และการออกแบบตัวควบคุมของตัวย่อยเยื่อกระดาษ. วิทยานิพนธ์ปริญญาโท, จุฬาลงกรณ์มหาวิทยาลัย.
- วฤต ศิลป์ศรีกุล. 2534. การออกแบบตัวควบคุมร่วมของระบบเชิงเส้นหลายระบบ โดยคำนึงถึงสมรรถนะแบบ H2 ด้วยวิธีอสมการเมทริกซ์เชิงเส้น. วิทยานิพนธ์ปริญญาโท, จุฬาลงกรณ์มหาวิทยาลัย.
- เดอะพีซีไกด์. 2547. **Hard Disk Read/Write Head Technologies Hard Disk Drive ปี 2547**. แหล่งที่มา: <http://www.pcguides.com/ref/hdd/op/over.htm>; 13 มิถุนายน 2552.
- รีแพร์โน้ตบุ๊กคอตเน็ต . 2009. **เทคนิคการกู้ข้อมูล ปี 2009**. แหล่งที่มา: <http://repair-notebook.com/2009/03/03.html>; 5 สิงหาคม 2552.

วิทยา พรพัชรพงศ์. 2551. โครงข่ายประสาทเทียม ปี 2551. แหล่งที่มา:

<http://gotoknow.org/blog/business-intelligence/163433>; 15 สิงหาคม 2553.

Ahmed, M.S. and I.A. Tasadduq. 1994. Neural network controller for linear plants design approach through linearization. **IEE Proc.-Control Theory Appl** 141: 315-322.

Akhyar, S. and S. Omatu. (1984). Neuromorphic Self-Tuning PID Controller. **In IEEE International Conference on Neural Networks** vol. 1: 552-557.

Ahn, K.K. and N.H.T.Chau. 2007. Design of a robust force controller for the new mini motion package using quantitative feedback theory. **ScienceDirect** 17: 542-550.

Chen, F.C. 1990. Back propagation neural networks for nonlinear self-tuning adaptive control. **IEEE Control System Magazine** 10: 44-48.

Cui, X. and K.G. Shin. 1993. Direct control and coordination using neural networks. **IEE Transaction on systems Man and Cybernetics** 23: 686-697.

Hung T. N. 2002. **A First Course in Fussy and Neural Control**.
CHAPMAN&HALL/CRC, New Mexico.

Irwin, G.W. and G. Lightbody. 1995. Direct neural model reference adaptive control. **IEE Proc-Control Theory Appl** 142: 31-43.

Jin, L., P.N. Nikiforuk, and M.M. Gupta. 1994. Adaptive control of discrete time nonlinear systems using neural networks. **IEE Proc-Control Theory Appl** 141: 169-176

Kundergi, R. and P.S.V.Nataraj. 1994. Evaporator Control Design: A Quantitative Feedback Theory Approach. **IEEE** 2: 1473-1474.

Leuba, C.M. and R.B. Perez. 1998. Optimal Control of Uncertain Nonlinear System: An Application to a Two-Link Robot Arm. **IEEE** 1: 123-129.

Narendra, K.S. and K. Parthasarathy. 1990. Identification and control of dynamic system using neural network. **IEEE Transaction on Neural Network** 1: 4-27.

Nordgern, R.E. and P.H. Meckel. 1993. An analytical comparison of neural network and a model based adaptive controller. **IEEE Transaction on Neural Network** 4: 685-694.

Sastry, P.s., G. Santharam, and K.P. Unnikrishnan. 1994. Memory neural network for identification and control of dynamic system. **IEEE Transaction on Neural Network** 5 : 306-319.

Weerasoriya, S. and E.L. Sharkawi. 1991. Identification and control of dc motor using back propagation neural networks. **IEEE Transaction on Energy Conversion** 6 : 663-669.

Yamamura, A., A. Sideris, A. Ji and C. Psaltis. 1990. Neural network control of a two-link manipulator. **IEEE** 6: 3265 – 3266.





โปรแกรมแมทแลปที่ใช้ในการออกแบบระบบควบคุมอัตโนมัติโดยอาศัยหลักการและทฤษฎีของโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ ประกอบด้วย 3 ขั้นตอนคือ

1. โครงข่ายประสาทเทียมที่ใช้ในการหาเอกลักษณ์

```
%-----
% MATLAB neural network model
% by Krittiya Pa-im
% 22 August 2011
%-----

close all;
clear all;
load Data3.mat;
clc;

%user specified values
hidden_neurons = 140; %hidden neural
epochs = 3000;
alpha = 0.01; %learning rate
gole = 1e-2;

% ----- load in the data -----

plant_inp = velocity;
plant_out = torque;
siz = size(velocity,1); siz_2 = siz-1;

train_inp = [plant_out(1:siz_2,:) plant_inp(2:siz,:)];
train_out = plant_out(2:siz,:);
```

```

% check same number of patterns in each
if size(train_inp,1) ~= size(train_out,1)
    disp('ERROR: data mismatch')
    return
end

%read how many inputs
inputs = size(train_inp,2);

% ----- set weights -----
%set initial random weights
wi = randn(hidden_neurons,inputs); %weight input hidden
wo = randn(hidden_neurons,1);      %weight hidden output

% ----- set bias -----
%set initial random bias
bi = randn(hidden_neurons,1); %bias input hidden
bo = randn(1);                %bias hidden output

%read how many patterns
patterns = size(train_inp,1);

%----- data loaded -----

%----- add some control buttons -----

%add button for early stopping
hstop = uicontrol('Style','PushButton','String','Stop', 'Position', [5 5 70 20],'callback','earlystop =
1;');
earlystop = 0;

```

```

%-----
%--- Learning Starts Here! -----
%-----

%do a number of epochs
for i = 1:epochs

    %loop through the patterns, selecting randomly
    for j = 1:patterns

        %set the current pattern
        inp = train_inp(j,:);
        out = train_out(j,1);

        %calculate the current error for this pattern
        tic;          %start timer
        z = logsig(wi*inp'+bi);
        y = wo'*z+bo;

        for k=1:hidden_neurons
            zz(k,k) = (1-z(k,1))*z(k,1);
        end

        sigmao = (out-y)*1;
        sigmah = zz*wo*sigmao;

        % adjust weight hidden - output
        delta_wo = alpha*sigmao*z;
        wo = wo + delta_wo;
    end
end

```

```

% adjust the weights input - hidden
delta_wi= alpha*sigmah*inp;
wi = wi + delta_wi;

% adjust bias hidden - output
delta_bo = alpha*sigmao;
bo = bo + delta_bo;

% adjust the bias input - hidden
delta_bi= alpha*sigmah;
bi = bi + delta_bi;

pred(j,:) = y;

Timer(j) = toc; %stop timer and get result

end

% -- another epoch finished

%plot overall network error at end of each epoch
error = train_out - pred;
err(i) = 0.5*sum(error.^2);

figure(1);
plot(err)
title('Performance')
xlabel('Epochs')
ylabel('Mean Squared Error (MSE)');
set(figure(1),'name','Performance of neural network model training','numbertitle','off');

%stop if requested

```



```

if earlystack
    fprintf('stopped at epoch: %d\n',i);
    break
end

%stop if error is small
if err(i) < gole
    fprintf('converged at epoch: %d\n',i);
    break
end
end

%-----FINISHED-----

```

2. โครงข่ายประสาทเทียมที่ใช้ในการควบคุม

```

%-----
% MATLAB neural network controller
% by Krittiya Pa-im
% 23 August 2011
%-----

```

```

close all;
clear all;
load after_train_model.mat
clc;

%user specified values
hidden_neurons = 70; %hidden neural
epochs = 100;

```

```

alpha = 0.01;      %learning rate
gole = 0.01;

% ----- load in the data -----

signal_inp = ref_out;
signal_ref_out = ref_out;

train_inp_c = [signal_inp(2:1001,:) signal_ref_out(1:1000,:)];
train_out_c = signal_ref_out(2:1001,:);
yref = signal_ref_out(1:1000,:);    %delay

% check same number of patterns in each
if size(train_inp_c,1) ~= size(train_out_c,1)
    disp('ERROR: data mismatch')
    return
end

%read how many inputs
inputs_c = size(train_inp_c,2);

% ----- set weights -----
%set initial random weights
wi_ctrl = randn(hidden_neurons,inputs_c); %weight input hidden
wo_ctrl = randn(hidden_neurons,1);    %weight hidden output

% ----- set bias -----
%set initial random bias
bi_ctrl = randn(hidden_neurons,1); %bias input hidden
bo_ctrl = randn(1);    %bias hidden output

```

```

%read how many patterns
patterns = size(train_inp_c,1);

%----- data loaded -----

%----- add some control buttons -----

%add button for early stopping
hstop = uicontrol('Style','PushButton','String','Stop', 'Position', [5 5 70 20],'callback','earlystop =
1;');
earlystop = 0;

%-----
%--- Learning Starts Here! -----
%-----

%do a number of epochs
for i = 1:epochs

    %loop through the patterns, selecting randomly
    for j = 1:patterns

        %set the current pattern
        inc = train_inp_c(j,:);
        ref_delay = yref(j,1);

        %calculate the current error for this pattern

        tic;          %start timer
        zc = logsig(wi_ctrl*inc+bi_ctrl);
        yc = wo_ctrl*zc+bo_ctrl;

        for k=1:hidden_neurons

```

```

    zzc(k,k) = ((1-zc(k,1))*zc(k,1));
end

inp = [yc ref_delay];
z = logsig(wi*inp'+bi);
y = (wo'*z+bo);

hidden_neurons_p = size(wi,1);

for l=1:hidden_neurons_p
    zz(1,l) = ((1-z(1,1))*z(1,1));
end
    ref = train_out_c(j,1);

    sigmao = (ref-y)*1;
    sigmah = zz*wo*sigmao;
    E0 = wi'*sigmah;
    yn = E0(1,1);

    sigmao_c = (yn)*1;
    sigmah_c = zzc*wo_ctrl*sigmao_c;

    % adjust weight hidden - output
    d_wo_ctrl = alpha*zc*sigmao_c;
    wo_ctrl = wo_ctrl + d_wo_ctrl;

    % adjust the weights input - hidden
    d_wi_ctrl = alpha*sigmah_c*inc;
    wi_ctrl = wi_ctrl + d_wi_ctrl;

    % adjust bias hidden - output

```

```

d_bo_ctrl = alpha*sigmao_c;
bo_ctrl = bo_ctrl + d_bo_ctrl;

% adjust the bias input - hidden
d_bi_ctrl= alpha*sigmah_c;
bi_ctrl = bi_ctrl + d_bi_ctrl;

ym(j) = y;

Timer(j) = toc; %stop timer and get result

end

% -- another epoch finished

%plot overall network error at end of each epoch
error_c = train_out_c - ym';
errc(i) = sum(0.5*(error_c.^2));

figure(1);
plot(errc)
title('Performance')
xlabel('Epochs')
ylabel('Mean Squared Error (MSE)');
set(figure(1),'name','Performance of neural network model training','numbertitle','off');

%stop if requested
if earlystack
    fprintf('stopped at epoch: %d\n',i);
    break
end

```

```

%stop if error is small
if errc(i) < gole
    fprintf('converged at epoch: %d\n',i);
    break
end
end
end

```

```

%-----FINISHED-----

```

3. การทดสอบระบบควบคุม

```

%-----
% MATLAB neural network controller
% by Krittiya Pa-im
% 27 August 2011
%-----

```

```

close all;
clear all;
load after_train_ctrl2.mat;
clc;

```

```

%user specified values
alpha = 0.01;      %learning rate
E1 = 0.01;
E2 = 0.01;
y_1 = 0;
ref = ref(1:1000);

```



```
% ----- load in the data -----
```

```
pattern = size(ref,1);      %
```

```
%-----
```

```
%--- Simulation Starts Here! -----
```

```
%-----
```

```
%do a number of size
```

```
for i = 1:pattern
```

```
    % go to NNCtrl part
```

```
    inp_c = [ref(i) y_1];    %
```

```
    zc = logsig(wi_ctrl*inp_c'+bi_ctrl);
```

```
    yc = wo_ctrl'*zc+bo_ctrl;
```

```
    hidden_neurons = size(wi_ctrl,1);
```

```
    for j=1:hidden_neurons
```

```
        zzc(j,j) = (1-zc(j,1))*zc(j,1);
```

```
    end
```

```
    uc(i,:) = yc;
```

```
        % go to Plant
```

```
        inp = [yc y_1];
```

```
        z = logsig((wi)*inp'+bi);
```

```
        y = ((wo)*z+bo);
```

```
    % go to NNModel part
```

```

inp_m = [yc y_1];
zm = logsig(wi*inp_m'+bi);
ym = (wo'*zm+bo);

yout(i,:) = y;  %%%%%%%%%%
y_1=y;

hidden_neurons_p = size(wi,1);

for k=1:hidden_neurons_p
    zzm(k,k) = (1-zm(k,1))*zm(k,1);
end

% check error between yout and ym
err1 = sqrt((y-ym)^2); %%%%%%%%%%
err11(i,:) = err1;

if err1 > E1
    sigmao = (y-ym)*1;
    sigmah = zzm*wo*sigmao;

    % adjust weight hidden - output
    delta_wo = alpha*sigmao*z;
    wo = wo + delta_wo;

    % adjust the weights input - hidden
    delta_wi= alpha*sigmah*inp;
    wi = wi + delta_wi;

    % adjust bias hidden - output
    delta_bo = alpha*sigmao;

```

```

bo = bo + delta_bo;

% adjust the bias input - hidden
delta_bi= alpha*sigmah;
bi = bi + delta_bi;
end

% check error between yout and refout
err2 = sqrt((ref(i)-y)^2); %%%%%%%%%%
err22(i,:) = err2;

if err2 > E2
    sigmao_m = (ref(i)-y)*1;
    sigmah_m = zzm*wo*sigmao_m;
    E0 = wi'*sigmah_m;

    yn = E0(1,1);
    sigmao_c = (yn)*1;
    sigmah_c = zzc*wo_ctrl*sigmao_c;

    % adjust weight hidden - output CTRL
    d_wo_ctrl = alpha*zc*sigmao_c;
    wo_ctrl = wo_ctrl + d_wo_ctrl;

    % adjust the weights input - hidden CTRL
    d_wi_ctrl= alpha*sigmah_c*inp_c;
    wi_ctrl = wi_ctrl + d_wi_ctrl;

    % adjust bias hidden - output CTRL
    d_bo_ctrl = alpha*sigmao_c;
    bo_ctrl = bo_ctrl + d_bo_ctrl;

```

```
% adjust the bias input - hidden CTRL
```

```
d_bi_ctrl= alpha*sigmah_c;
```

```
bi_ctrl = bi_ctrl + d_bi_ctrl;
```

```
end
```

```
po(i,:) = [ref(i,:) yout(i,:)];
```

```
figure(1);
```

```
plot(po);
```

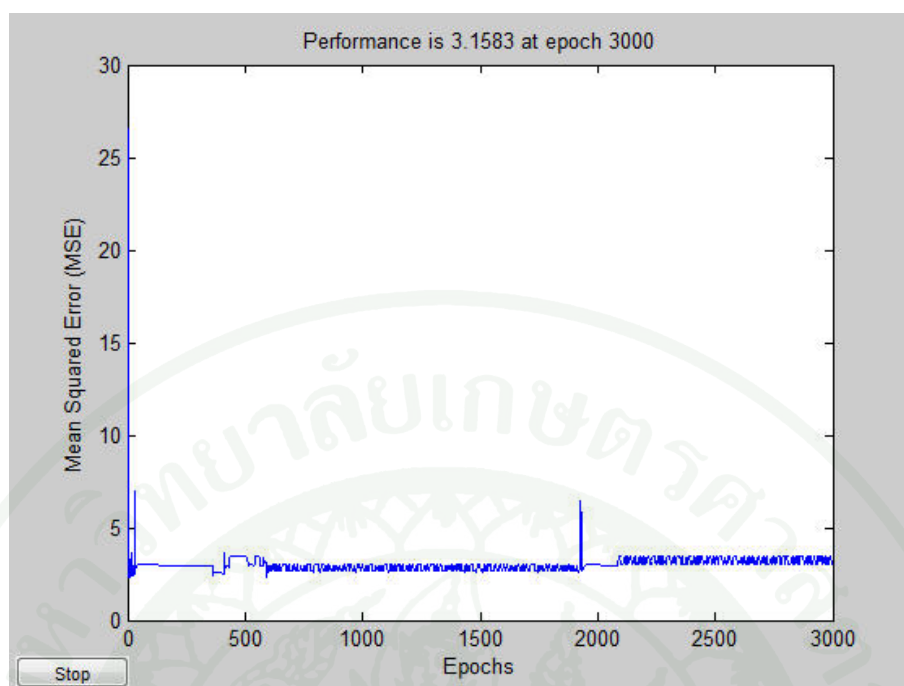
```
set(figure(1),'name','Plant Response for Neural Network Model','numbertitle','off');
```

```
end
```

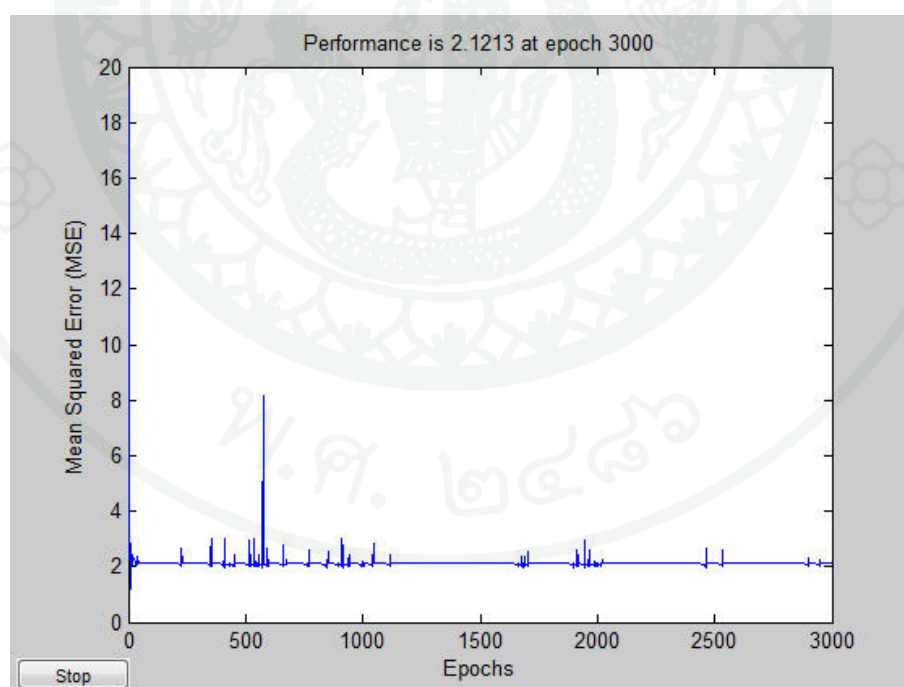


ภาคผนวก ข

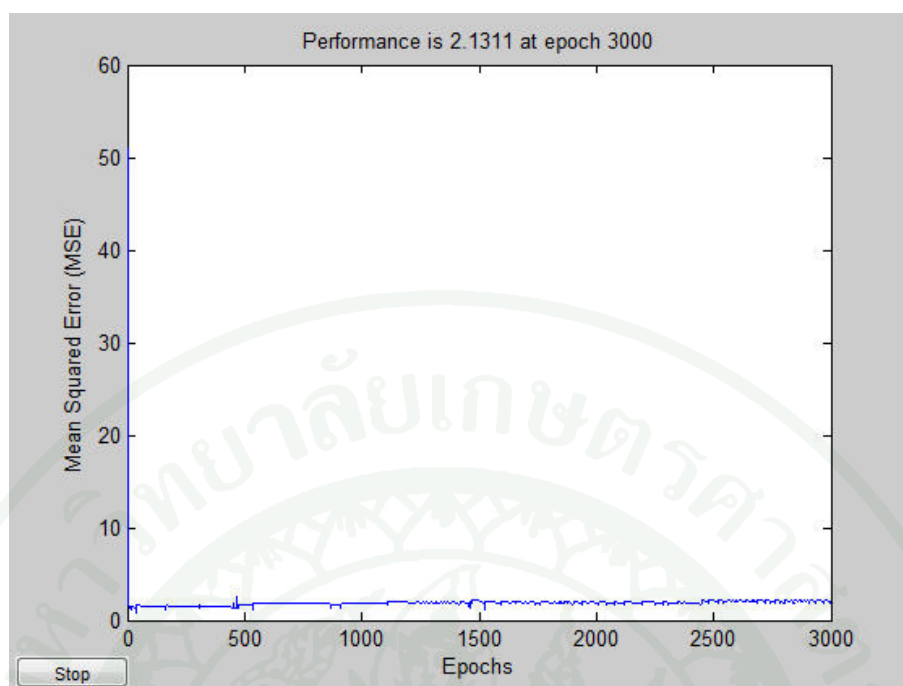
กราฟแสดงการเรียนรู้ของโครงสร้างของโครงข่ายประสาทเทียมในการหาเอกลักษณ์



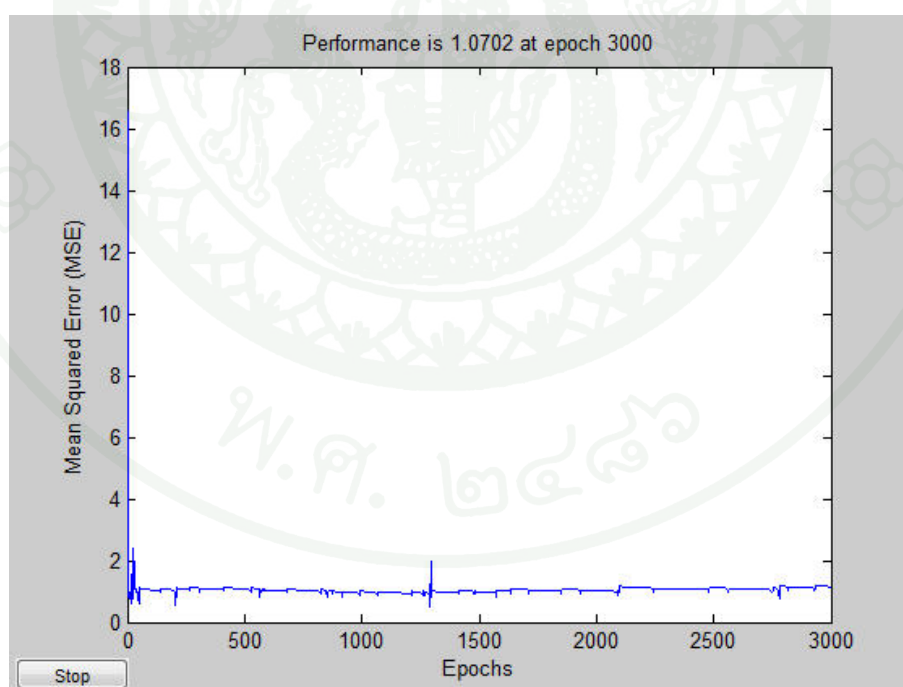
ภาพผนวกที่ ข1 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 1



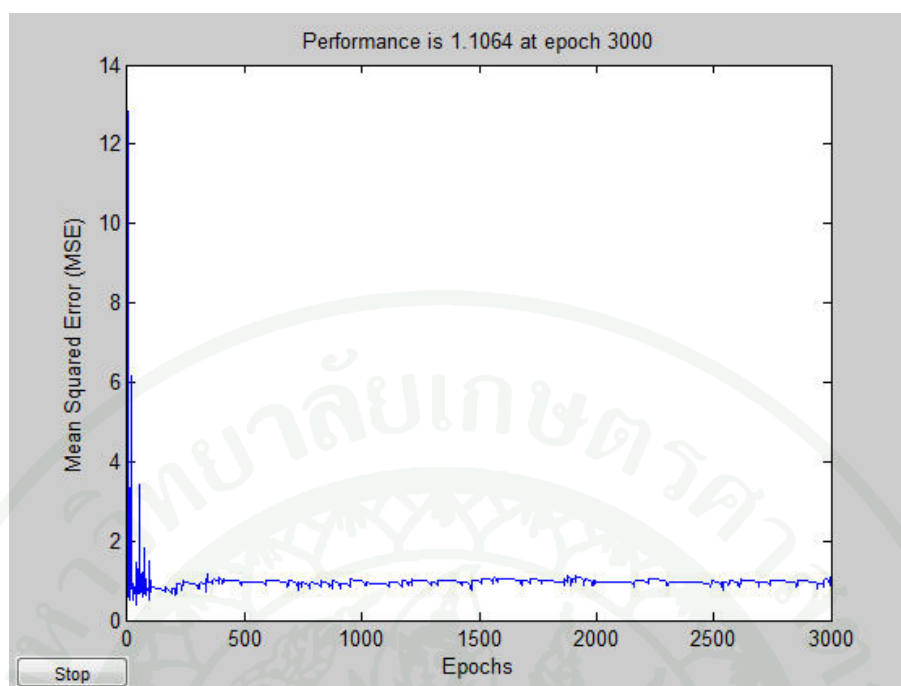
ภาพผนวกที่ ข2 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 2



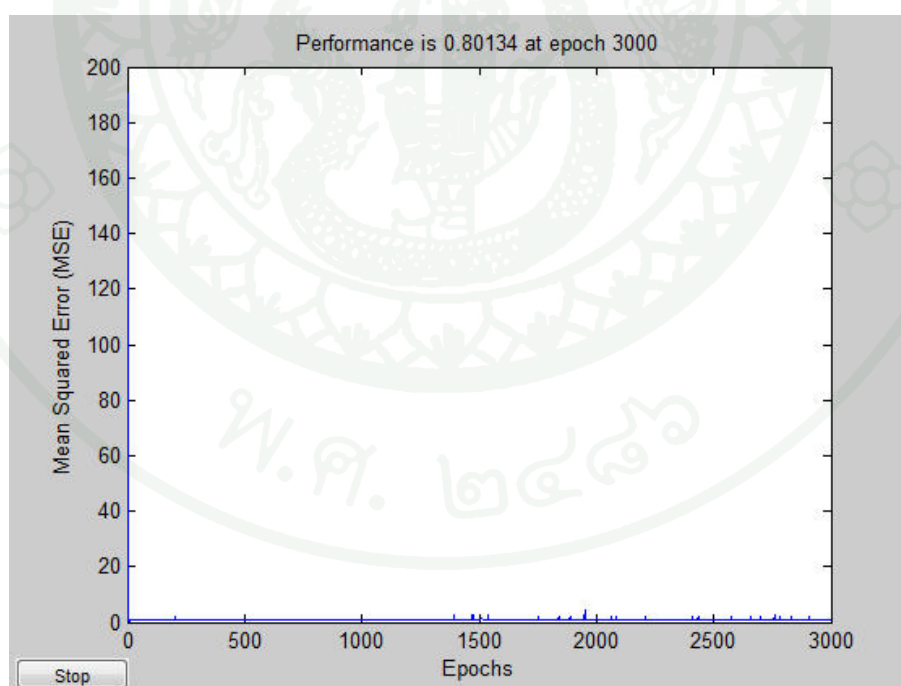
ภาพผนวกที่ ข3 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 3



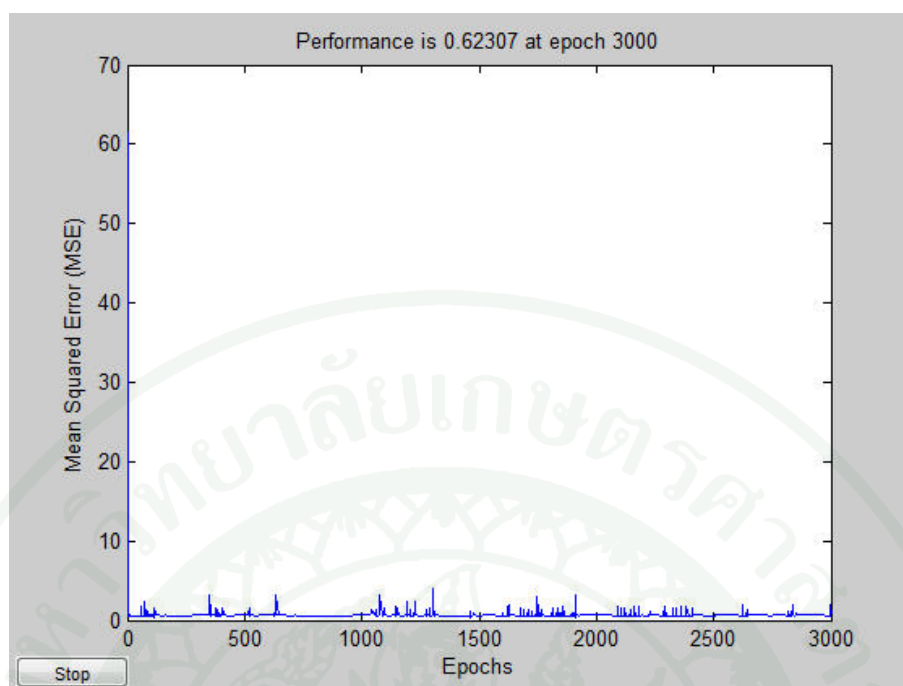
ภาพผนวกที่ ข4 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 4



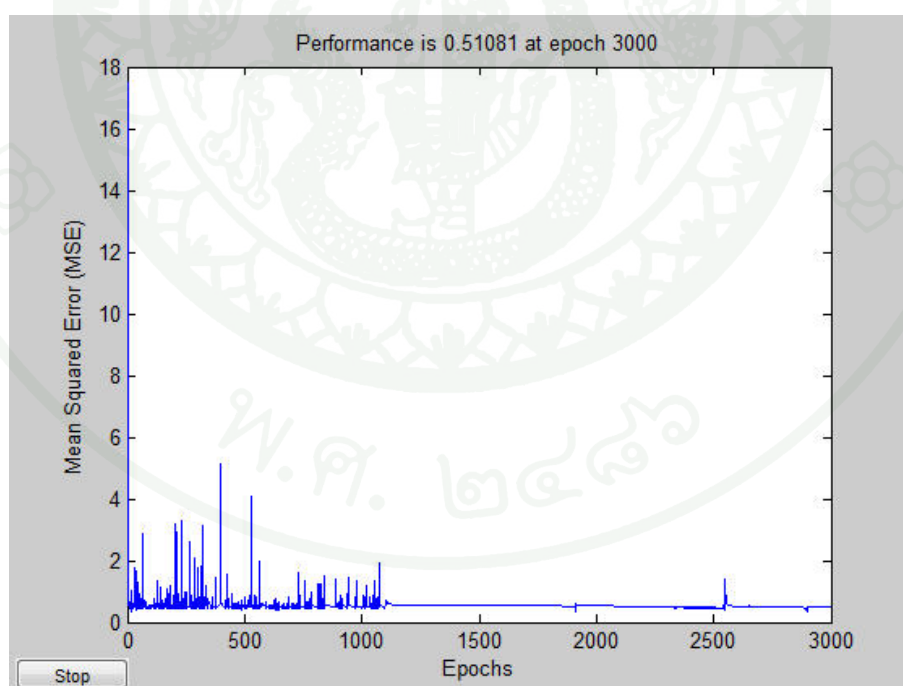
ภาพผนวกที่ ข5 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 5



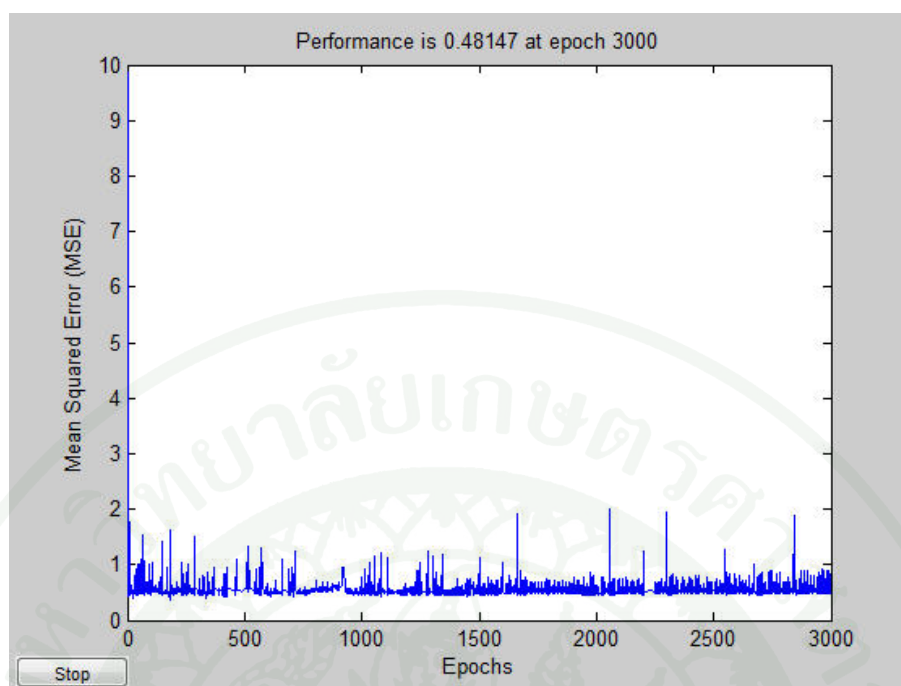
ภาพผนวกที่ ข6 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 6



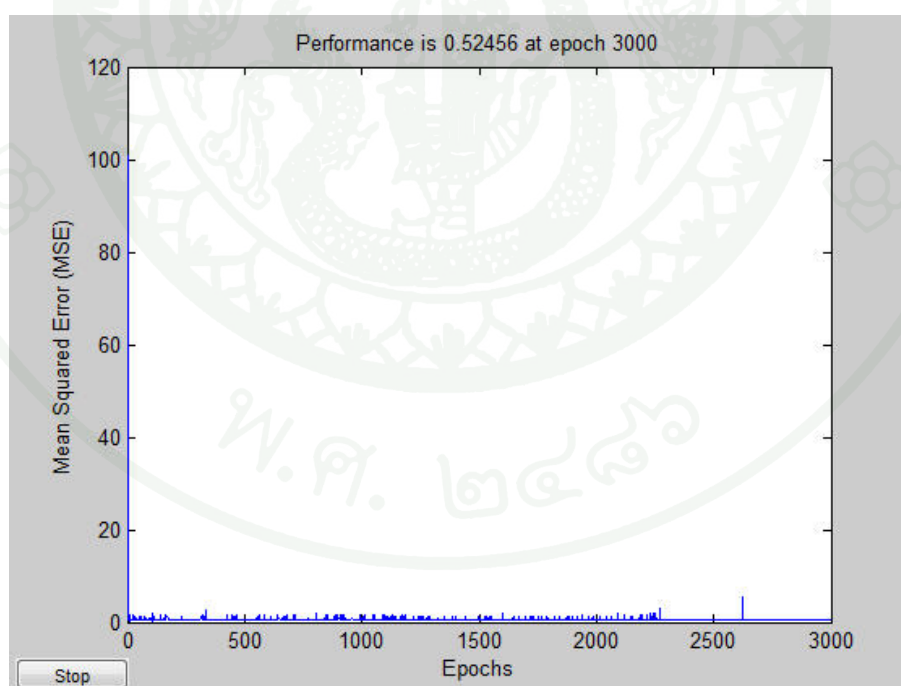
ภาพผนวกที่ ข7 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 7



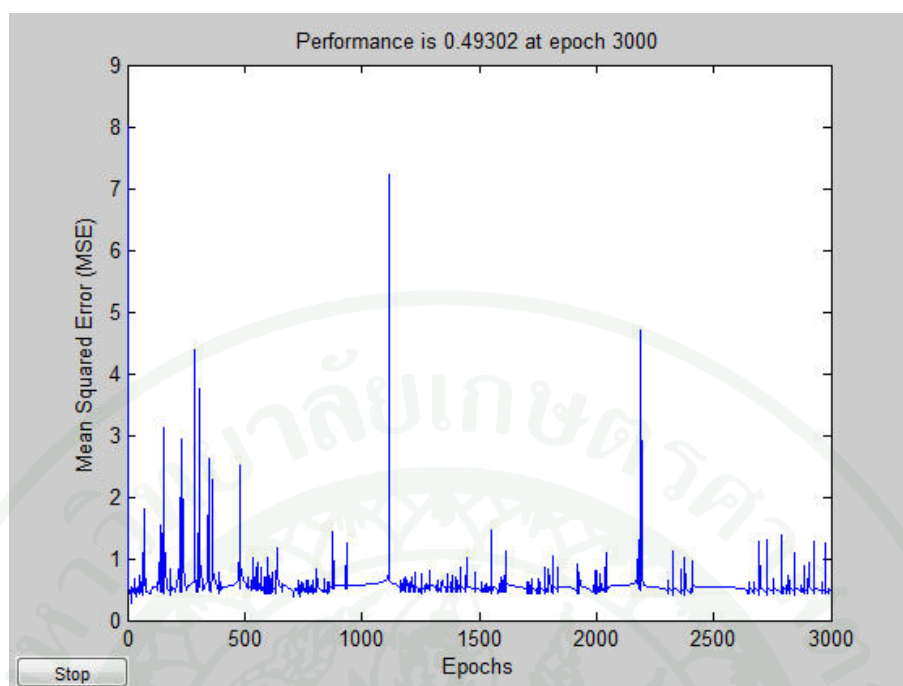
ภาพผนวกที่ ข8 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 8



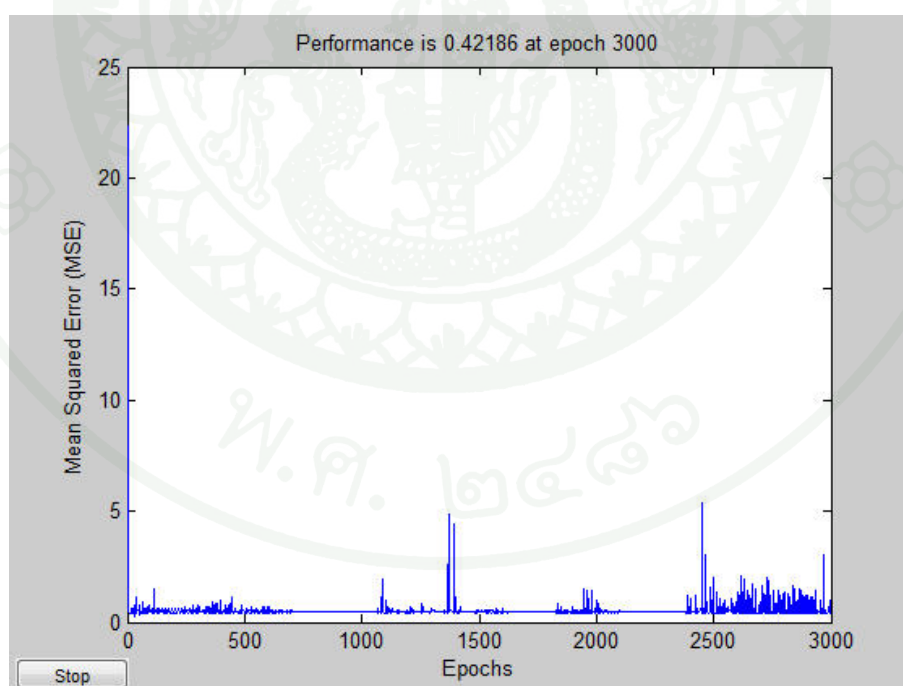
ภาพผนวกที่ ข9 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 9



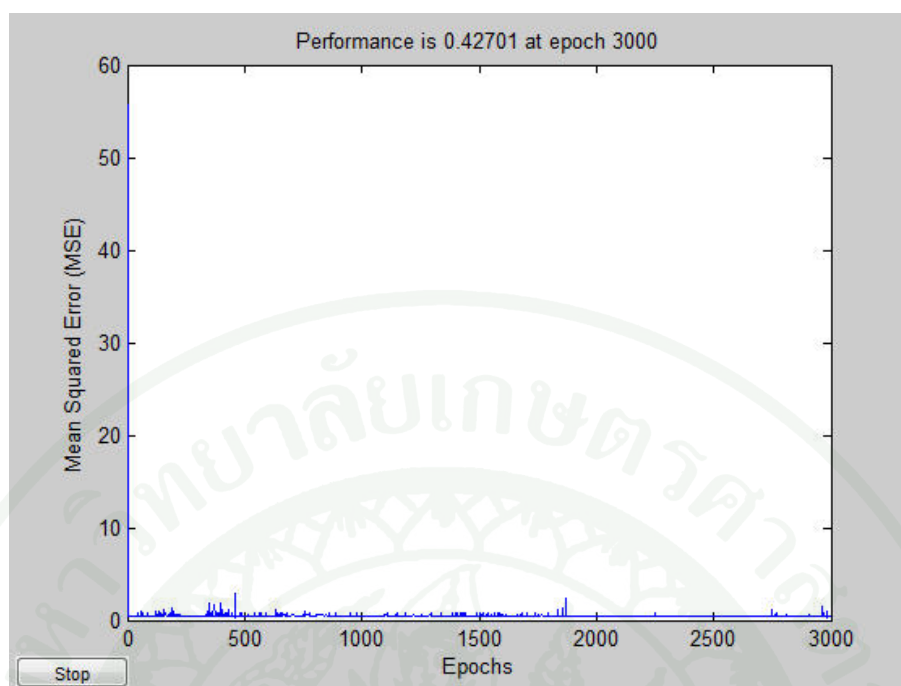
ภาพผนวกที่ ข10 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 10



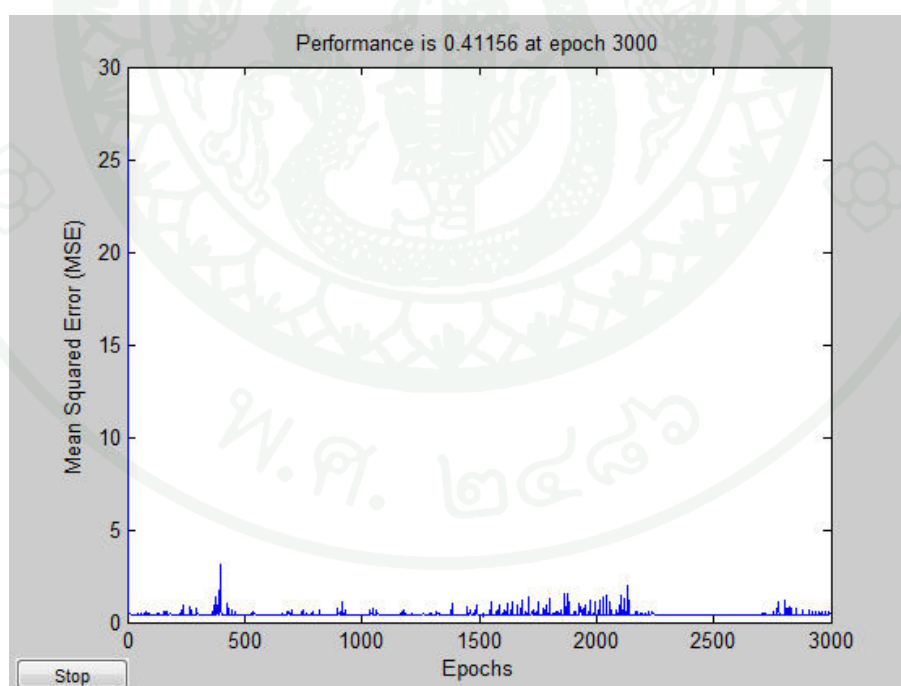
ภาพผนวกที่ ข11 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 11



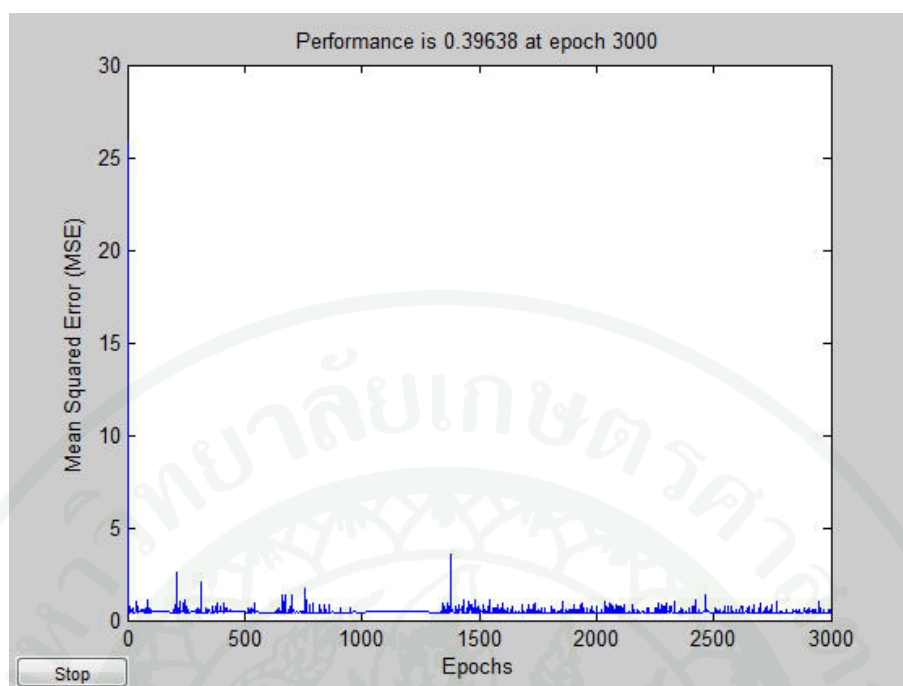
ภาพผนวกที่ ข12 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 12



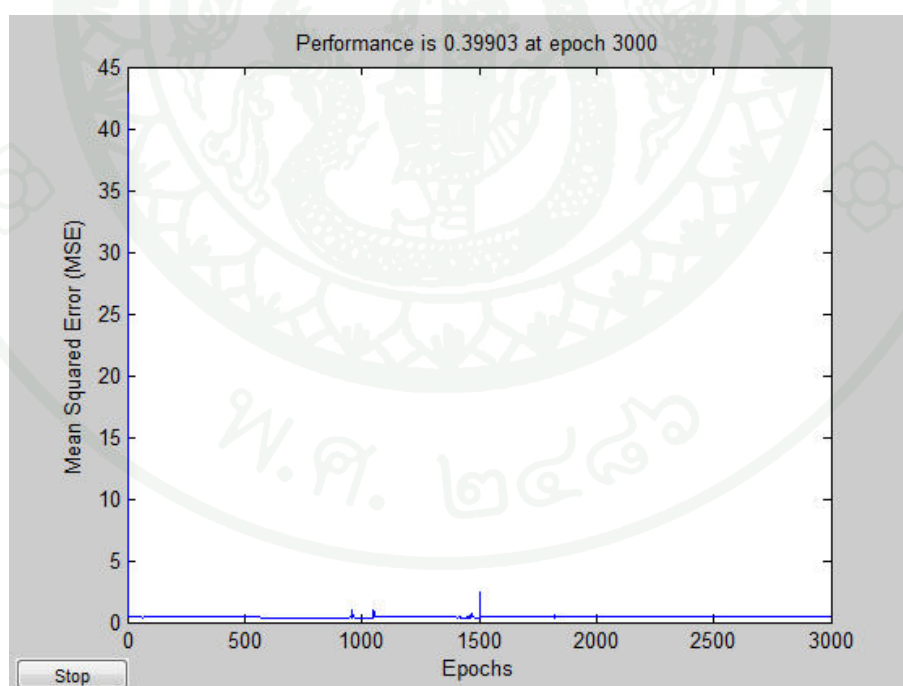
ภาพผนวกที่ ข13 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 13



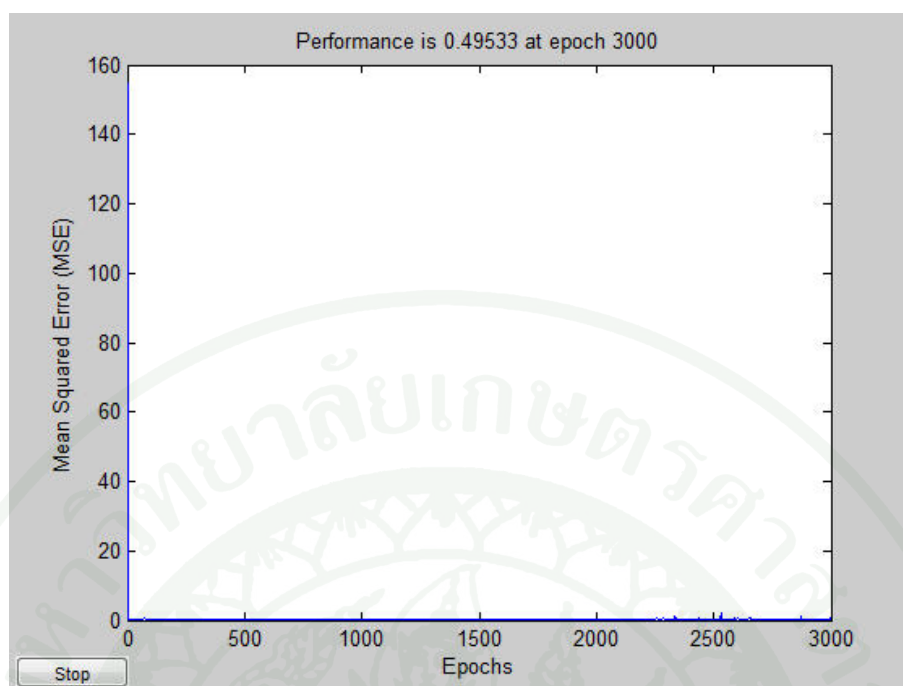
ภาพผนวกที่ ข14 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 14



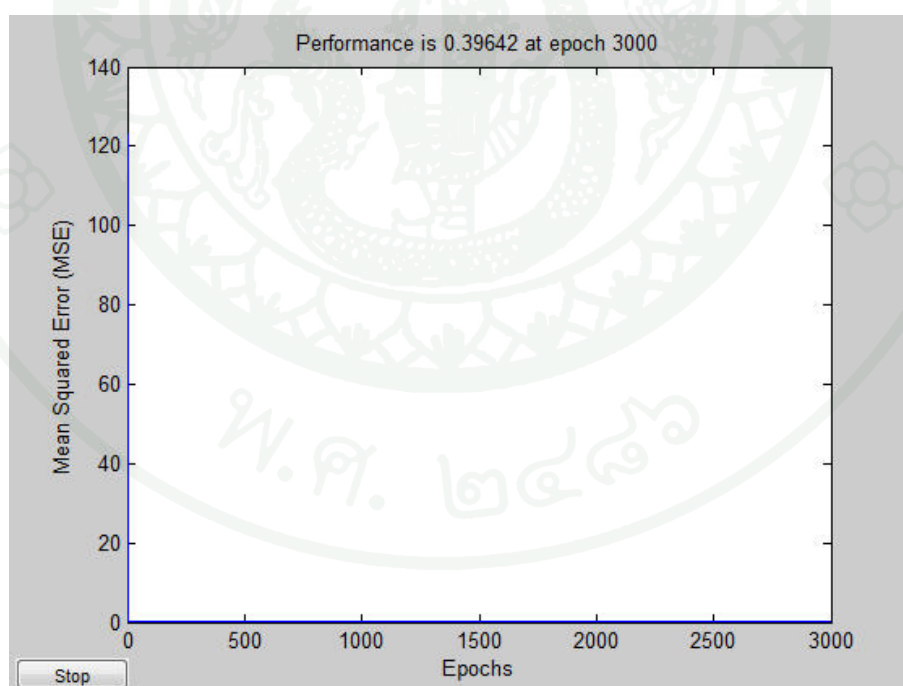
ภาพผนวกที่ ข15 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 15



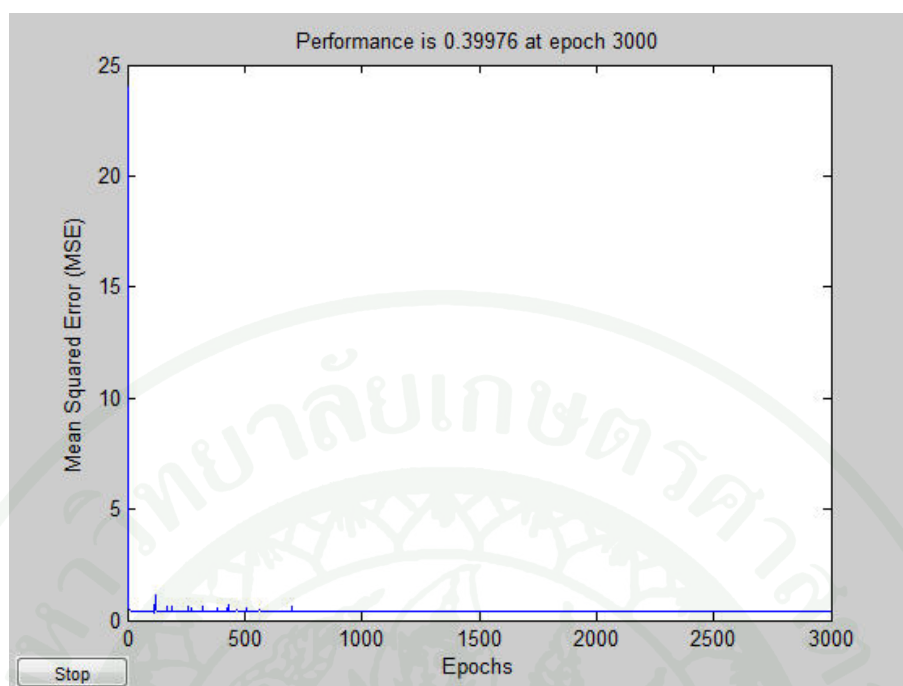
ภาพผนวกที่ ข16 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 16



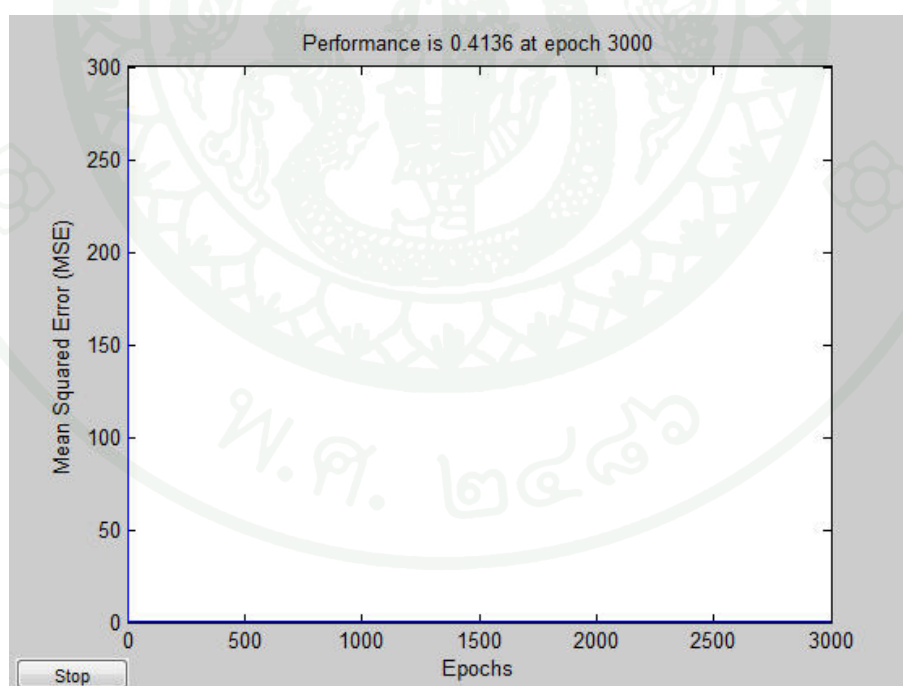
ภาพผนวกที่ ข17 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 17



ภาพผนวกที่ ข18 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 18



ภาพผนวกที่ ข19 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 19

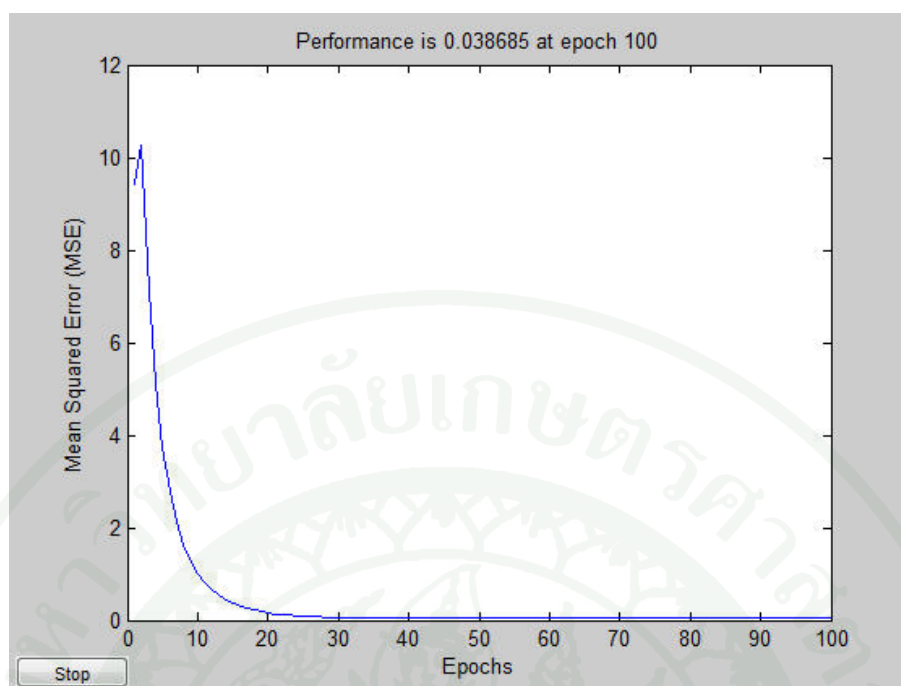


ภาพผนวกที่ ข20 กราฟแสดงการเรียนรู้ของโครงสร้างในการหาเอกลักษณ์รูปแบบที่ 20

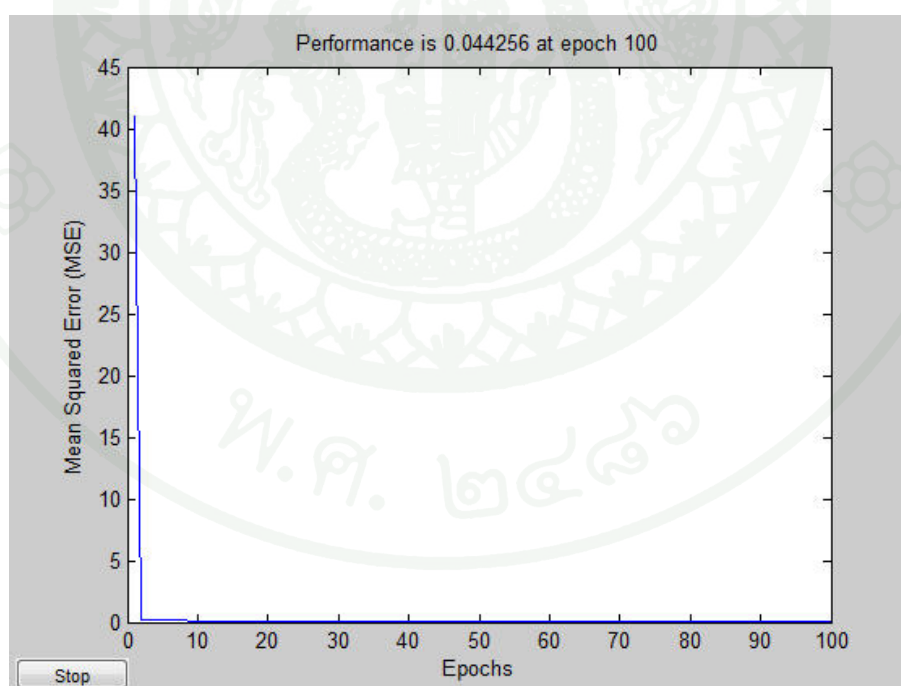


ภาคผนวก ค

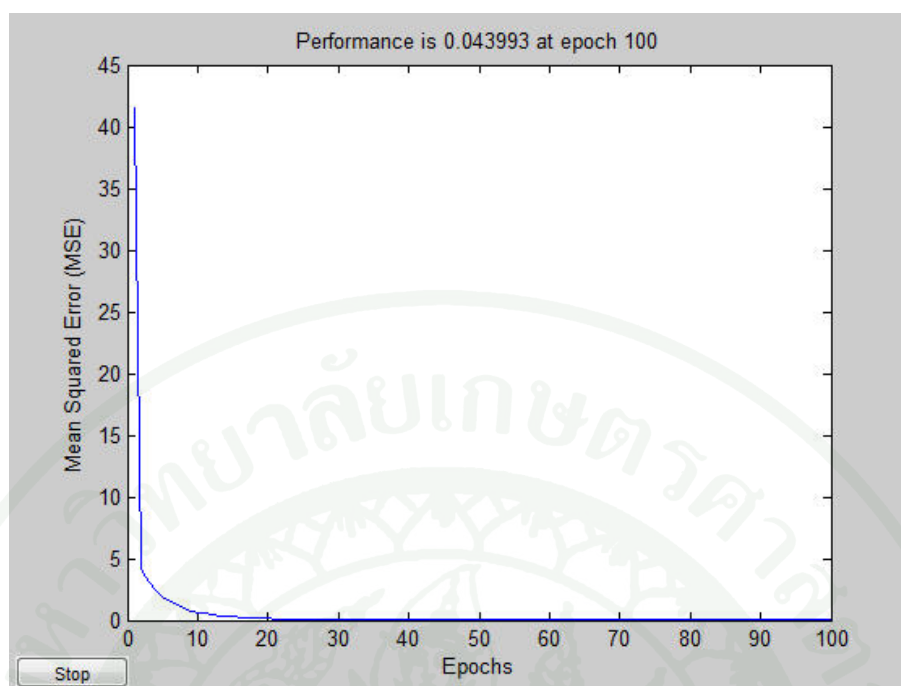
กราฟแสดงการเรียนรู้ของโครงสร้างของโครงข่ายประสาทเทียมในการควบคุม



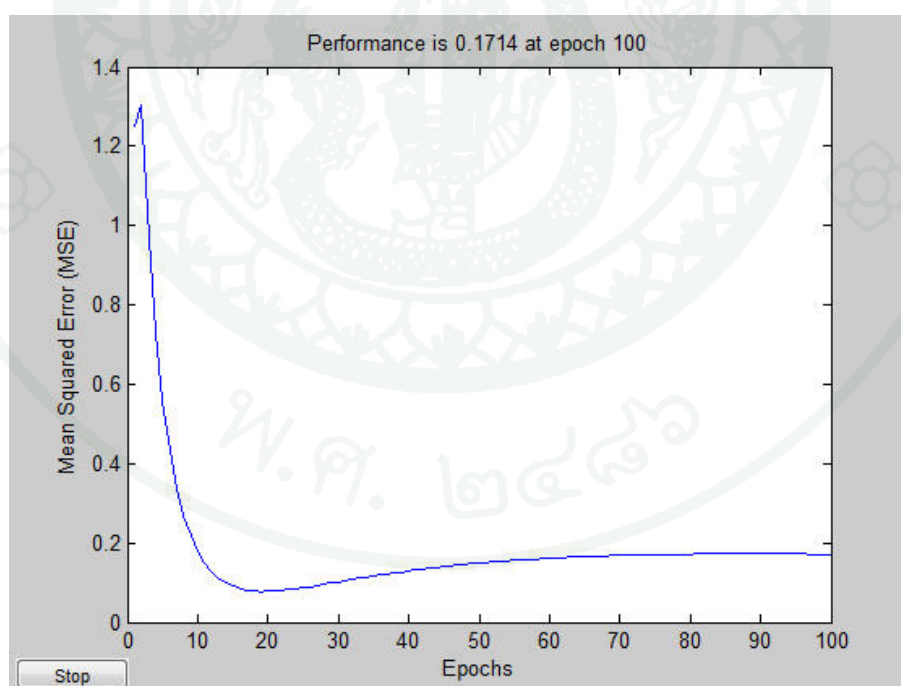
ภาพผนวกที่ ค1 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 1



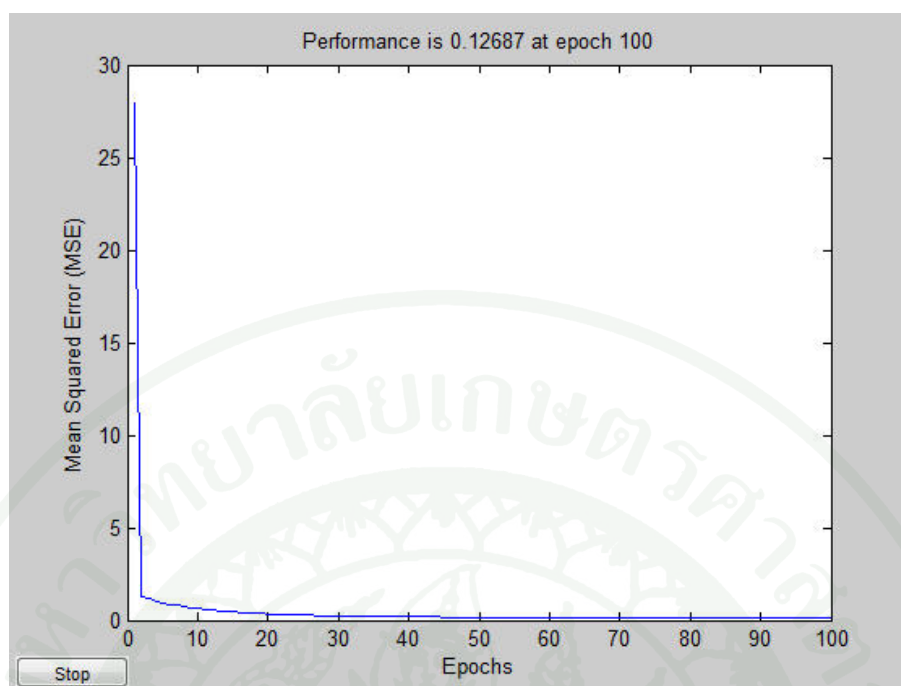
ภาพผนวกที่ ค2 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 2



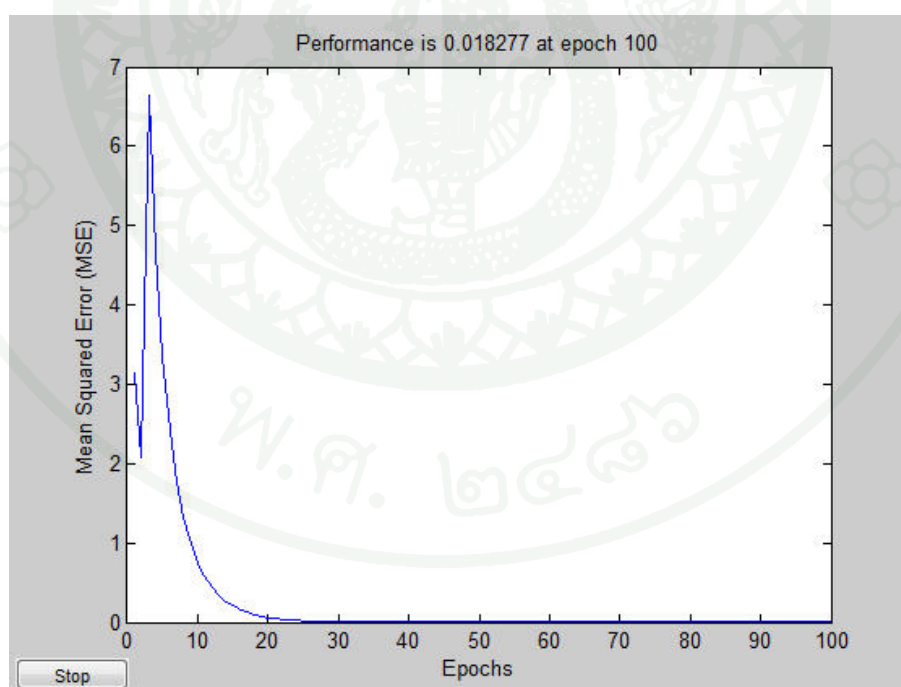
ภาพผนวกที่ ๓ กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 3



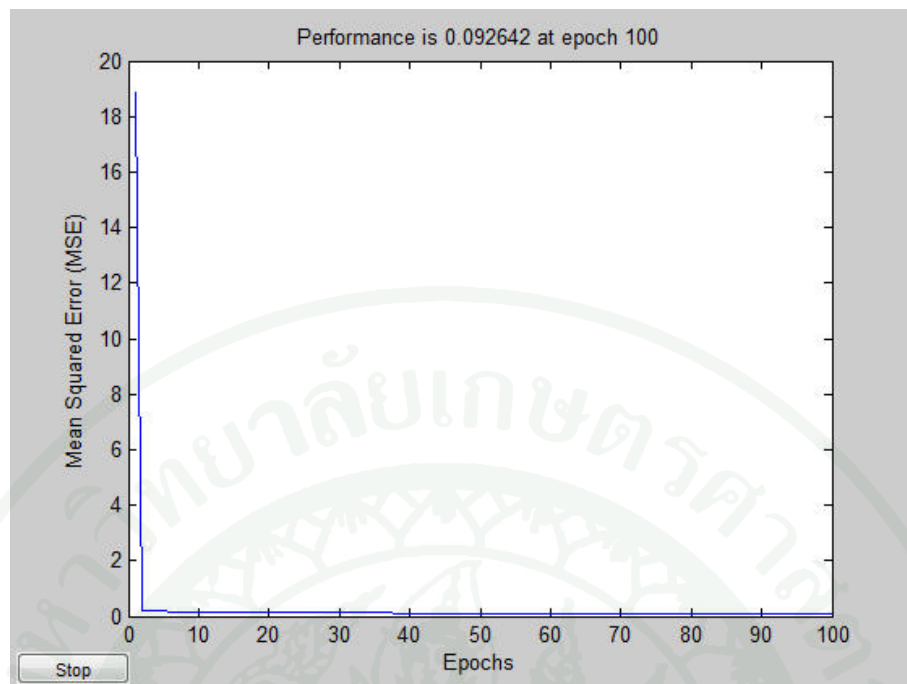
ภาพผนวกที่ ๔ กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 4



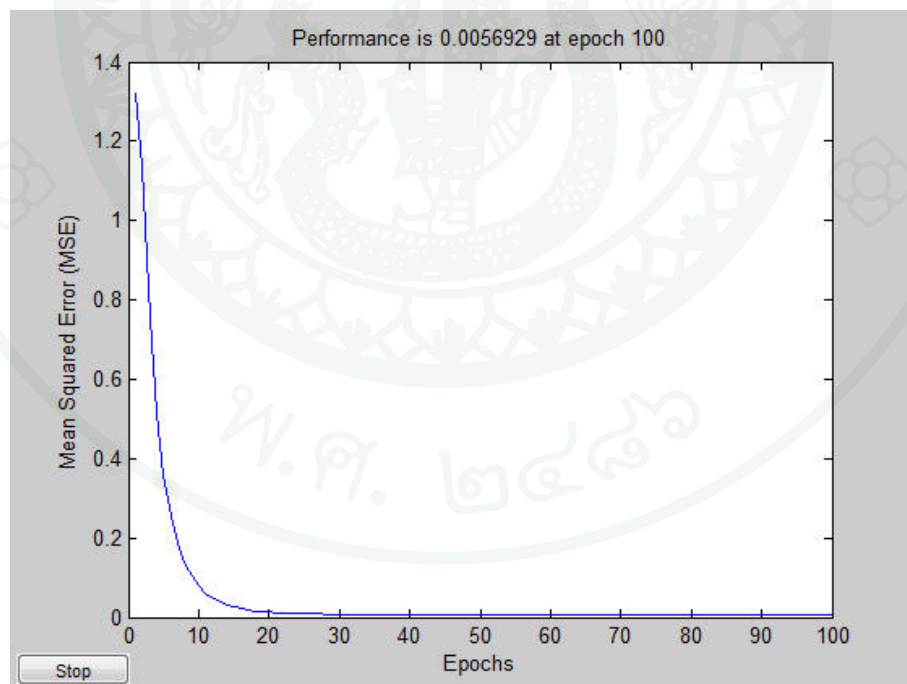
ภาพผนวกที่ ค5 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 5



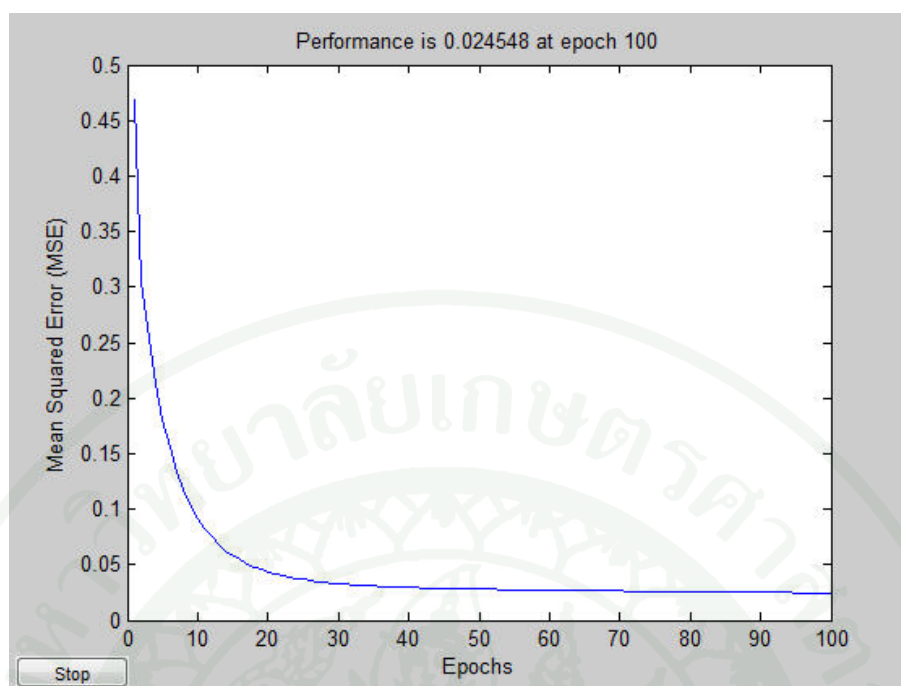
ภาพผนวกที่ ค6 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 6



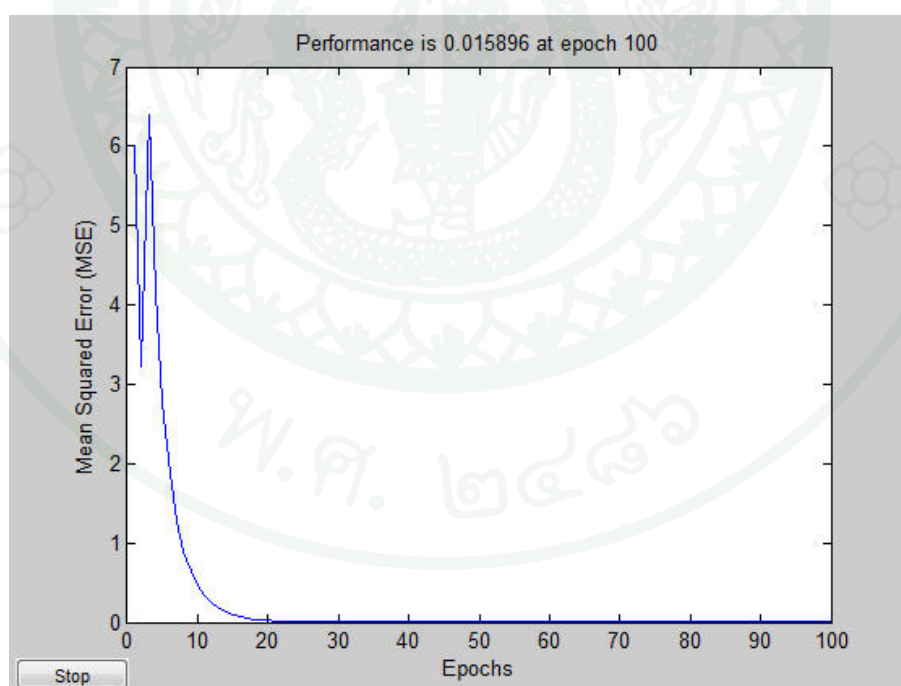
ภาพผนวกที่ ค7 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 7



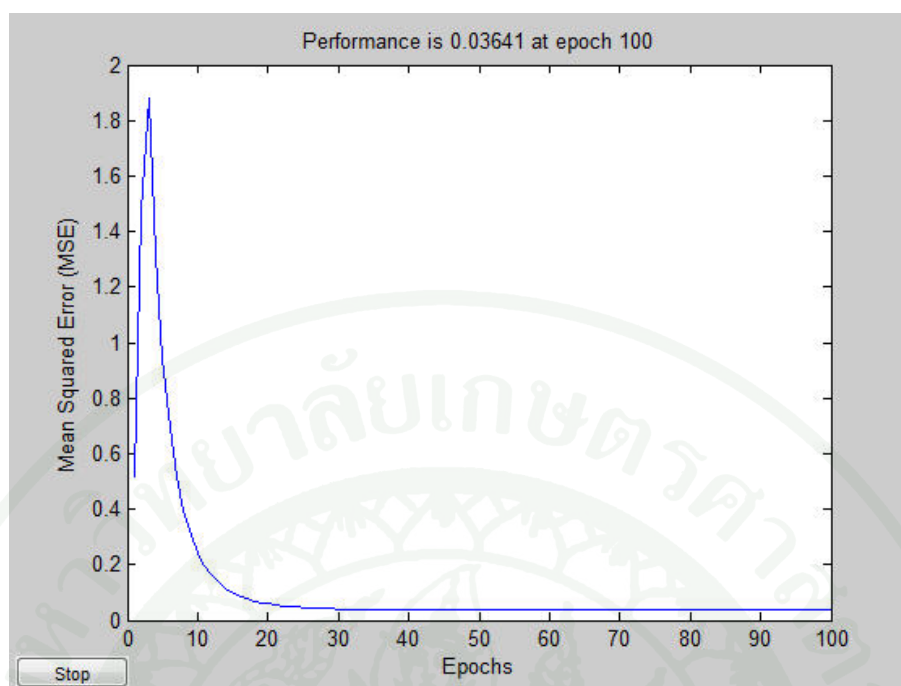
ภาพผนวกที่ ค8 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 8



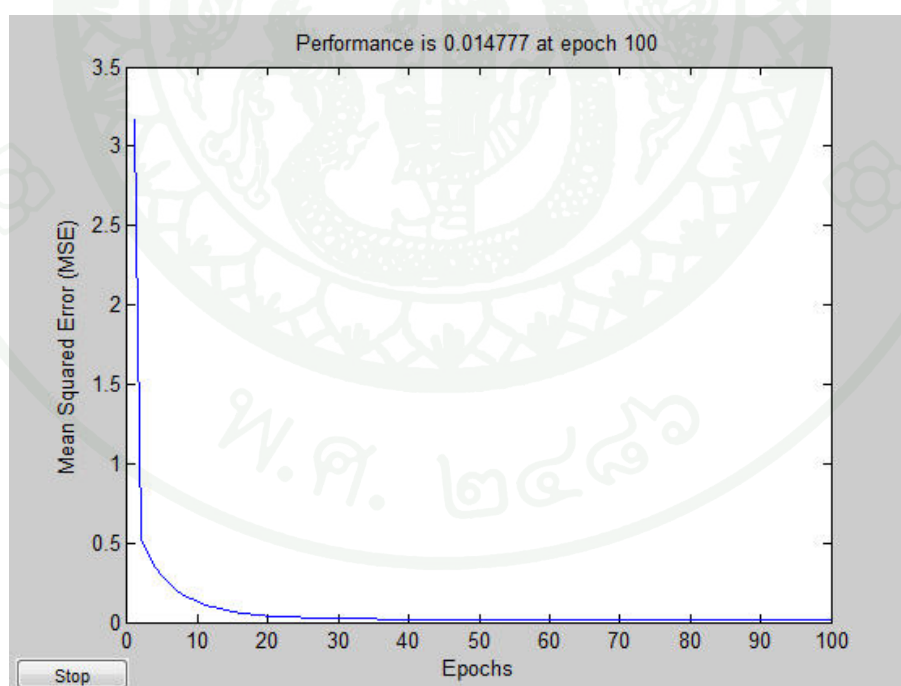
ภาพผนวกที่ ๙ กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 9



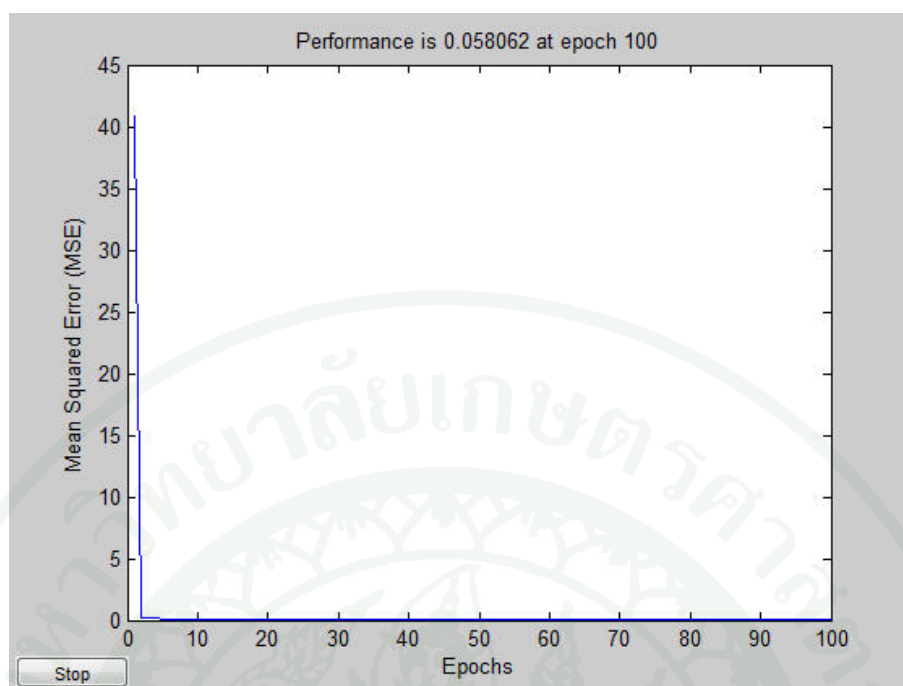
ภาพผนวกที่ ๑๐ กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 10



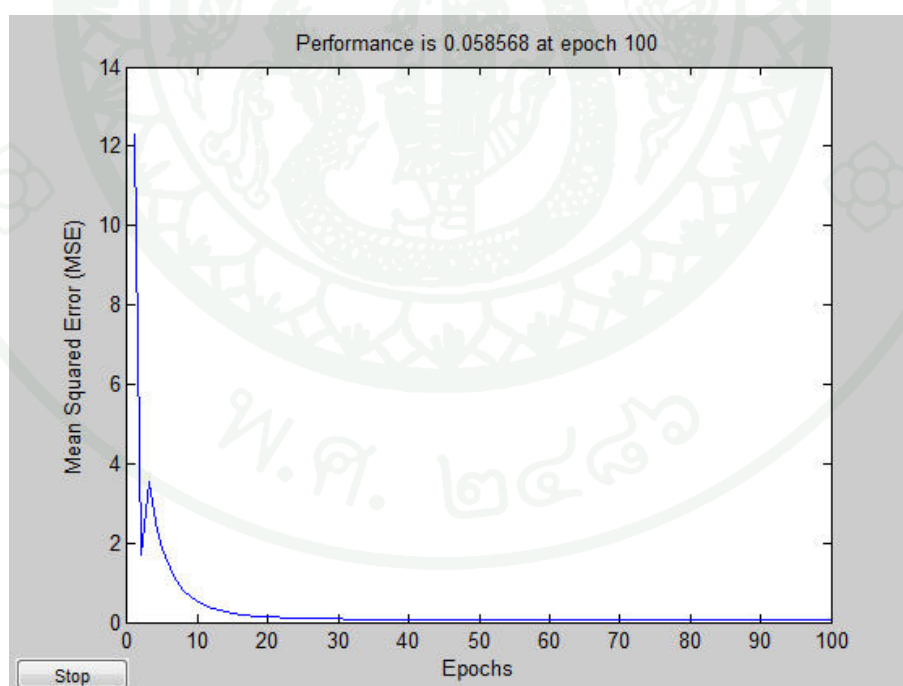
ภาพผนวกที่ ค11 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 11



ภาพผนวกที่ ค12 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 12



ภาพผนวกที่ ค13 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 13



ภาพผนวกที่ ค14 กราฟแสดงการเรียนรู้ของโครงสร้างในการควบคุมรูปแบบที่ 14

ประวัติการศึกษา และการทำงาน

ชื่อ นางสาวกฤติยา พาอ้อม
เกิดวันที่ 6 เมษายน 2529
สถานที่เกิด กรุงเทพมหานคร
ประวัติการศึกษา วศ.บ. (วิศวกรรมไฟฟ้าเครื่องกลการผลิต) มหาวิทยาลัยเกษตรศาสตร์
ตำแหน่งปัจจุบัน —
สถานที่ทำงานปัจจุบัน —
ผลงานดีเด่นและ/หรือรางวัลวิชาการ —
ทุนการศึกษาที่ได้รับ —