



รายงานวิจัยฉบับสมบูรณ์

โครงการ การใช้ Soft - Computing ในการสร้างรูปแบบการเดิน แบบพลวัตของหุ่นยนต์ฮิวแมนนอยด์

โดย

รศ.ดร. ชิต	เหล่าวัฒนา	หัวหน้าโครงการ
นายหทัยเทพ	วงศ์สุวรรณ	ผู้ช่วยนักวิจัย

สถาบันวิทยาการหุ่นยนต์ภาคสนาม (FIBO)
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

มีนาคม 2548

รายงานวิจัยฉบับสมบูรณ์

โครงการ

การใช้ Soft - Computing ในการสร้างรูปแบบการเดิน
แบบพลวัตของหุ่นยนต์ฮิวแมนนอยด์

โดย

รศ.ดร. ชิต	เหล่าวัฒนา	หัวหน้าโครงการ
นายหทัยเทพ	วงศ์สุวรรณ	ผู้ช่วยนักวิจัย

สถาบันวิทยาการหุ่นยนต์ภาคสนาม (FIBO)
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

สนับสนุนโดยสำนักงานกองทุนสนับสนุนการวิจัย (TRF)

(ความเห็นในรายงานฉบับนี้เป็นของผู้วิจัย สกว. ไม่จำเป็นต้องเห็นด้วยเสมอไป)

กิตติกรรมประกาศ

งานวิจัยนี้ได้เงินทุนสนับสนุนจากทุนองค์ความรู้ใหม่สำหรับผลิตนักวิจัยปริญญาเอก(คปก) และ ทุนปริญญาเอกกาญจนาภิเษก(คปก) สำนักงานกองทุนสนับสนุนการวิจัย(สกว) โดยถือเป็นส่วนหนึ่งของการศึกษาระดับปริญญาเอกที่ สถาบันวิทยาการหุ่นยนต์ภาคสนาม (FIBO) มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี (KMUTT) ขอขอบคุณอาจารย์ที่ปรึกษา รศ. ดร. ชิต เหล่าวัฒนา ที่ให้การสนับสนุนและให้คำแนะนำตลอดมา นอกจากนี้ขอขอบคุณผลงานการออกแบบและสร้างหุ่นยนต์ฮิวแมนนอยด์ของนักศึกษาปริญญาโท นายไพศาล สุวรรณเทพ นายจิตติศักดิ์ จันทร์พรหม นายบัณฑิต ศรีสุวรรณ โครงการวิจัย Image Guided Motion of a Humanoid Robot ที่ได้รับทุนสนับสนุนจากศูนย์อิเล็กทรอนิกส์ และคอมพิวเตอร์แห่งชาติ (Nectec) ที่ใช้เป็นต้นแบบในการทดลอง

Project Code : BGJ4380028

Project Title : Soft-computing – Gait generation for dynamic walking of a Humanoid Robot

Investigator : Hataitep Wongsuwarn, Assoc. Prof. Djitt Laowattana

E-mail Address : hataitep@fibo.kmutt.ac.th, Djitt@fibo.kmutt.ac.th

Project Period : 30 September 2001 – 29 August 2003

Abstract

During the last few years, the number of research and development projects of bipedal and humanoid robots has been increasing at a rapid rate. In this research, we provide a brief review of current activities in the field of bipedal and humanoid robots. We particularly emphasize the use of biologically inspired computation methods in field of humanoid robotics. Firstly, we discuss some intelligent control techniques (neural networks, fuzzy logic and genetic algorithms) and their hybrid methods (neuro-fuzzy networks, neuro-genetic and fuzzy-genetic and neural-fuzzy-genetic algorithms), focusing on system identification and modeling. Secondly, we introduce an intelligent control technique. Third, we develop a bipedal gait synthesis technique based on an adaptive neuro-fuzzy controller. We developed an architecture which generates trunk trajectory in sagittal plane and lateral plane. Our proposed gait synthesizer was trained through sets of a Zero Moment Point (ZMP). Furthermore, we experimented on an 8-link biped robot walking in sagittal plane and lateral plane. The robot is both headless and armless. The hierarchical control software for the biped robot was developed using MATLAB, SIMULINK, Fuzzy Logic Toolbox and Neural Network Toolbox. The main contribution of this research is a general neuro-fuzzy synthesis of using gait generating algorithm. This algorithm implementation leads to any dynamic bipedal walking without requiring for dynamic equations and nominal data.

Keywords Soft-computing, Neuro-fuzzy modeling and control, Humanoid robot, Biped robot

รหัสโครงการ : BGJ4380028

ชื่อโครงการ : การใช้การคำนวณเชิงอ่อนในการสร้างรูปแบบการเดินแบบพลวัตของหุ่นยนต์ฮิวแมนนอยด์

ชื่อนักวิจัย : นายหทัยเทพ วงศ์สุวรรณ, รศ. ดร. ชิต เหล่าวัฒนา

E-mail Address: hataitep@fibo.kmutt.ac.th, Djitt@fibo.kmutt.ac.th

ระยะเวลาโครงการ : 30 กันยายน 2543 ถึง 29 สิงหาคม 2545

บทคัดย่อ

หลายๆปีที่ผ่านมางานวิจัยและการพัฒนาหุ่นยนต์สองขาและหุ่นยนต์ฮิวแมนนอยด์ได้เพิ่มมากขึ้นอย่างรวดเร็ว ในงานวิจัยนี้กล่าวถึงงานวิจัยด้านหุ่นยนต์สองขาและฮิวแมนนอยด์ในอดีต และการใช้ระบบการคำนวณเชิงชีวภาพหรือระบบอัจฉริยะที่ใช้ในงานหุ่นยนต์สองขาและฮิวแมนนอยด์ โดยเริ่มต้นบรรยายถึงเทคนิคระบบอัจฉริยะต่างๆเช่นข่ายงานระบบประสาท ตรรกศาสตร์คลุมเครือ กลวิธีทางพันธุกรรม และระบบรวมเช่น ระบบนิวโรฟัชชี ระบบนิวโรเจเนอติก อื่นๆ ที่ใช้ในการจำลองระบบทางกายภาพต่างๆ จากนั้นได้บรรยายถึงการพัฒนาแบบสังเคราะห์ท่าทางการเดินของหุ่นยนต์โดยใช้ระบบนิวโรฟัชชี โดยระบบดังกล่าวนี้ใช้ลำตัวและหลักการโมเมนตัมรวมศูนย์เป็นเกณฑ์ในการออกแบบ และการทดลองใช้หุ่นยนต์สองขาขนาดเล็กที่มี 8 แกน โดยระบบควบคุมถูกพัฒนาโดยใช้โปรแกรม MATLAB, SIMULINK, Fuzzy Logic Toolbox และ Neural Network Toolbox. จากการทดลองพบว่าระบบสามารถเรียนรู้และปรับตัวให้หุ่นยนต์สามารถเดินได้อย่างดี ผลของงานวิจัยนี้ทำให้ได้ระบบการเรียนรู้เพื่อสร้างท่าทางการเดินของหุ่นยนต์ โดยไม่จำเป็นต้องมีสมการทางพลวัตของหุ่นยนต์ แต่อาศัยเพียงข้อมูลที่ได้จากหุ่นยนต์เท่านั้น

คำสำคัญ ระบบคำนวณเชิงอ่อน, ระบบนิวโรฟัชชี, หุ่นยนต์ฮิวแมนนอยด์, หุ่นยนต์สองขา

สารบัญ

กิตติกรรมประกาศ

บทคัดย่อภาษาอังกฤษ

บทคัดย่อภาษาไทย

บทที่ 1 บทนำ	1
บทที่ 2 ระบบการคำนวณเชิงอ่อน	6
บทที่ 3 การออกแบบระบบนิวโรฟัซซี่	23
บทที่ 4 หุ่นยนต์สองขาที่ใช้ในการทดลอง	33
บทที่ 5 การทดลอง	50
บทที่ 6 สรุปและวิจารณ์ผล	63
หนังสืออ้างอิง	65
Output จากโครงการวิจัยที่ได้ทุนจากสกว	69

บทที่ 1 บทนำ

1.1 ความเป็นมา

การขยายตัวของอุตสาหกรรมมีแนวโน้มเพิ่มขึ้นเรื่อยๆ เพื่อให้การผลิตเป็นไปอย่างมีประสิทธิภาพ อุตสาหกรรมการผลิตแบบอัตโนมัติจึงเข้ามามีบทบาทในวงการอุตสาหกรรม เทคโนโลยีหุ่นยนต์ (Robotic technology) ซึ่งเป็นส่วนหนึ่งของอุตสาหกรรมการผลิตแบบอัตโนมัติได้ถูกนำมาช่วยในการผลิตโดยหุ่นยนต์ส่วนใหญ่เป็นแขนกลชนิดยึดอยู่กับที่ (Fixed Robot) แต่เนื่องจากการผลิตบางประเภทต้องการการขนย้าย เช่น การขนถ่ายวัตถุดิบที่มีความร้อนสูง การนำวัตถุดิบเข้าสู่สายการผลิต หรือแม้กระทั่งการสำรวจในพื้นที่อันตรายเป็นต้นทำให้หุ่นยนต์แบบเคลื่อนที่ (Mobile Robot) ถูกนำมาใช้กับงานดังกล่าว

โดยทั่วไปหุ่นยนต์แบบเคลื่อนที่ที่นิยมนำมาใช้เป็นแบบหุ่นยนต์แบบเคลื่อนที่ด้วยล้อ (Wheeled robot) และหุ่นยนต์แบบเคลื่อนที่ด้วยขา (Walking Robot) หุ่นยนต์แบบเคลื่อนที่ด้วยล้อที่ใช้กันอย่างแพร่หลายมีตั้งแต่ล้อเดี่ยวไปจนถึงหลายล้อเพื่อใช้ในการขับเคลื่อนให้หุ่นยนต์ไปยังตำแหน่งและทิศทางที่ต้องการ โดยทั่วไปหุ่นยนต์แบบล้อสามารถเคลื่อนที่ในพื้นที่ที่ค่อนข้างราบเรียบได้ดี แต่หุ่นยนต์แบบล้อไม่เหมาะสมกับภูมิประเทศที่สลับซับซ้อนและขรุขระ ดังนั้นหุ่นยนต์ที่เคลื่อนที่ด้วยขาจึงมีความเหมาะสมกับภูมิประเทศดังกล่าวมากกว่าหุ่นยนต์ที่เคลื่อนที่ด้วยล้อ โดยสามารถเคลื่อนที่ได้ในภูมิประเทศที่มีความขรุขระ พื้นที่ต่างๆ หุ่นยนต์ที่เคลื่อนที่ด้วยขาที่ทำการวิจัยมาตั้งแต่อดีตจนถึงปัจจุบันมีจำนวนตั้งแต่หนึ่งขาเป็นต้นไป ยิ่งจำนวนขาน้อยยิ่งส่งผลให้ความคล่องตัวในการเคลื่อนที่มีมากขึ้นเท่านั้น หุ่นยนต์ที่เคลื่อนที่ด้วยขาสองขา (Humanoid Robot) มีความสามารถในการเคลื่อนที่ (Mobility) ได้ดีกว่าหุ่นยนต์ที่เคลื่อนที่ด้วยขาแบบอื่น แต่เนื่องจากหุ่นยนต์สองขามีส่วนประกอบของโครงสร้างหลายชิ้นส่วนจึงทำให้การควบคุมให้หุ่นยนต์มีเสถียรภาพทั้งขณะที่อยู่กับที่และขณะก้าวเดินเป็นเรื่องที่ค่อนข้างยากและสลับซับซ้อนดังนั้นจึงได้มีการทำวิจัยการควบคุมดังกล่าว

การเดินของมนุษย์นั้นจะเห็นว่ามีกระบวนการเรียนรู้ลักษณะการเดินมาตั้งแต่เด็กในตอนที่เด็กหัดยืนนั้นตัวเด็กเองจะล้มบ่อยมาก ซึ่งเป็นผลมาจากตัวเด็กเองไม่สามารถที่รักษาสมดุลตัวเองให้อยู่ในลักษณะที่ตั้งฉากกับพื้นได้ แต่เมื่อเวลาผ่านไปนับจากวันที่หัดยืนซึ่งเห็นว่าเด็กสามารถที่รักษาสมดุลตัวเองไม่ให้ล้มได้ ซึ่งเป็นผลมาจากการที่เด็กได้เรียนรู้ความผิดพลาดในการยืน และได้เก็บการเรียนรู้เหล่านั้นเป็นข้อมูลในการปรับปรุงการทรงตัวของกายในภายหลัง การเรียนรู้ในลักษณะนี้คงมีไปเรื่อยแม้กระทั่งเด็กนั้นโตขึ้น แต่ลักษณะการเรียนรู้เหล่านั้นนั้นผิดแปลกแตกต่างกันขึ้นอยู่กับสภาพแวดล้อมที่พบ เช่นการเรียนรู้การเดินบนผิวราบกับผิวที่มีความลาดเอียง ซึ่งเห็นว่าการเรียนรู้เหล่านั้นมีส่วนสำคัญในการเดินของมนุษย์ จากที่กล่าวมานั้นการที่เราสร้างหุ่นยนต์ให้สามารถเดินได้เหมือนกับมนุษย์นั้น การสร้างการสมดุลในตัวหุ่นยนต์จึงเป็นเรื่องที่สำคัญมาก นอกจากนี้การควบคุมหุ่นยนต์สองขานั้นเป็นเรื่องทำได้ยาก [1,2] เนื่องจากเหตุผลหลายประการ อาทิเช่นความไม่เป็นเชิงเส้นของระบบพลวัตของตัวหุ่น อิทธิพลของแรงหนีศูนย์กลาง สภาพแวดล้อมที่กระทำต่อตัวหุ่นไม่ว่าเป็นสภาพพื้นผิวที่เปลี่ยนแปลงไป สิ่งกีดขวางต่างๆ ระบบเป็นแบบหลายสัญญาณเข้า และหลายสัญญาณออก (Multi Input -Multi Output) ระบบที่

เปลี่ยนแปลงตามเวลา ตัวอย่างเช่น Single Support Double support ballistic phase เป็นต้น นอกจากนี้ยังที่เกี่ยวกับการควบคุมแบบต่อเนื่อง (Continuous control) และแบบไม่ต่อเนื่อง (Discrete control) สำหรับการเคลื่อนที่แบบทีละก้าว (Step to step transition) จากการศึกษาถึงงานวิจัยในอดีตจึงมีผู้วิจัยหุ่นยนต์สองขาที่ออกไป เป็นหมวดๆ อันได้แก่

1. การศึกษาถึงกลไกในการออกแบบหุ่นยนต์สองขาให้สามารถเคลื่อนที่ได้ดีในทุกๆสถานที่
2. การศึกษาและออกแบบระบบควบคุมเสถียรภาพของข้อต่อหุ่นยนต์สองขา ซึ่งถือว่าเป็นระบบที่มีองศาอิสระสูง มีความไม่เป็นเชิงเส้นในระบบสูง และยังคงคำนึงถึงสภาวะสมดุลย์ในการเคลื่อนที่แบบสถิตย์และพลวัต
3. การออกแบบรูปแบบการเดินของหุ่นยนต์สองขาให้เคลื่อนที่ได้ในสถานที่ต่าง ๆ รวมถึงการวิเคราะห์หารูปแบบการเดินที่เหมาะสมและประหยัดพลังงานมากที่สุดอีกด้วย

นอกจากนี้ ในการทราบถึงการควบคุมระบบของหุ่นยนต์ที่ดีนั้น เราใช้วิธีการหลายแบบในการวัดสมรรถนะของระบบหุ่นยนต์สองขา อาทิเช่น ความคล้ายคลึงกับคน (Biological Similarity) ประสิทธิภาพของการเคลื่อนที่ในเชิงพลังงาน (Efficiency of Distance traveled per unit energy) การเคลื่อนที่ที่ราบเรียบ (Locomotion Smoothness) ความเร็วสูงสุดในการเดิน (Top Speed) ความสามารถในการเดินในพื้นที่ที่ไม่ราบเรียบ (Robustness to rough terrain)

ทางสถาบันวิทยาการหุ่นยนต์ภาคสนาม(FIBO) มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ได้ตระหนักและเล็งเห็นถึงความสำคัญของเทคโนโลยีที่ประเทศไทยสามารถพัฒนาขึ้นจากการทำวิจัยด้าน Humanoid Robot โดยทางสถาบัน FIBO ได้มีแผนการวิจัยและสร้างหุ่นยนต์ต้นแบบสองขาขึ้น สำหรับงานวิจัยนี้มุ่งเน้นไปที่หัวข้อที่ 2 และ 3 ที่เกี่ยวข้องกับการออกแบบรูปแบบการเดินของหุ่นยนต์(Gait Generation)ที่หุ่นยนต์สามารถเดินได้อย่างสมดุลย์และประหยัดพลังงานให้มากที่สุด โดยใช้กลวิธีการทางปัญญาประดิษฐ์ (Artificial Intelligence) มาช่วยให้หุ่นยนต์สามารถเคลื่อนที่ได้อย่างมีประสิทธิภาพมากขึ้น หุ่นยนต์เดินสองขาได้มีการศึกษาโดยนักวิจัยมาเป็นเวลาสามสิบปี เรื่องราวต่างๆของการพัฒนาหุ่นยนต์สองขาได้ถูกพูดถึงในบทความของ Todd[3] และ Raibert[4] สำหรับแนวทางในการนำระบบควบคุมแบบดั้งเดิม(Conventional control approach) มาใช้ในการควบคุมการเคลื่อนที่ได้จากบทความของ Sano[5] และ Furusho[6] สำหรับ Benbrahim และ Franklin[7] ได้นำวิธีการเรียนรู้แบบเสริมเข้มแข็ง(Reinforce Learning) มาใช้ในการควบคุมการเคลื่อนที่แบบพลวัตของหุ่นยนต์สองขา(Biped Dynamic Balance) Miller และคณะ[8] ได้รวมสองวิธีการเรียนรู้เข้าด้วยกันคือการเรียนรู้แบบมีผู้ช่วยสอน(Supervised Learning) และการเรียนรู้แบบผลต่างชั่วคราว(Temporal difference learning) มาใช้ในการหารูปแบบการเดินที่ต่อเนื่อง(Online gait adaptation) ซึ่งเป็นการจำลองความเป็นไปได้โดยใช้คอมพิวเตอร์สำหรับหุ่นยนต์เดินสองขาที่มีมวลขาน้อยๆ นอกจากนี้ยังได้มีการประยุกต์ใช้ข่ายงานระบบประสาทแบบ Radial Basis Function และ CMAC ในการหาแบบจำลองทางคณิตศาสตร์ที่ไม่เป็นเชิงเส้นของหุ่นยนต์เดินสองขาเพื่อให้สามารถเดินในที่ที่สภาวะแวดล้อมเปลี่ยนแปลงไปตามบทความของ Hu และ Pratt[9,10]

จากบทความที่ได้กล่าวมาข้างต้นเป็นการกล่าวถึงเฉพาะในแง่ของการนำข่ายงานระบบประสาทซึ่งมีความเหมาะสมในแง่ของความถูกต้อง แต่ใช้เวลาในการเรียนรู้มาก ซึ่งเป็นอุปสรรคต่อการควบคุมแบบทันที (Real-time control) ดังนั้นเพื่อแก้ปัญหาดังกล่าวจึงได้มีผู้คิดค้นวิธีการใหม่ขึ้นมา เรียกว่า “Soft-Computing” ซึ่งเป็นการนำเอาข้อดีของข่ายงานระบบประสาท (Neural Network) ระบบตรรกศาสตร์แบบคลุมเครือ (Fuzzy Logic System) และกลวิธีการถ่ายทอดพันธุกรรม (Genetic Algorithm) สิ่งที่ได้คือความสามารถในการควบคุมระบบซึ่งมีความไม่แน่นอนสูงได้ดีกว่าการควบคุมดั้งเดิม (Classical Control Approach) ตัวอย่างเช่น ระบบที่มีตัวแปรไม่ทราบค่าจำนวนมาก สรุปคือเราสามารถนำเอา Soft-Computing มาใช้ในการควบคุมหุ่นยนต์สองขา เพื่อให้สามารถเดินทรงตัวแบบพลวัติในสภาวะแวดล้อมต่างๆที่รู้ค่าและไม่รู้ค่าได้[11,12]

โดยในงานวิจัยนี้เป็นงานวิจัยพัฒนาทางทฤษฎีและตรวจสอบความถูกต้องจากหุ่นยนต์ต้นแบบจริง ซึ่งได้รับการสนับสนุนทุนวิจัยและสร้างหุ่นยนต์ต้นแบบจริงจากศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) โดยนำปัญญาประดิษฐ์แบบที่เรียกว่า Soft-computing ที่เป็นการรวมกันระหว่างหลักการของข่ายงานระบบประสาท (Neural Network) ระบบตรรกศาสตร์แบบคลุมเครือ (Fuzzy Logic System) และกลวิธีการถ่ายทอดพันธุกรรม (Genetic Algorithm) มาออกแบบลักษณะการเดินของหุ่นยนต์สองขาทั้งแบบสถิตย์ (Static) และแบบพลวัติ (Dynamic) ให้สามารถเดินได้ในพื้นที่สลับซับซ้อนและไม่มีการคาดการณ์ล่วงหน้า

1.2 วัตถุประสงค์/เป้าหมายของงานวิจัย

เพื่อออกแบบรูปแบบการเดินของหุ่นยนต์ (Gait Generation) สามารถควบคุมหุ่นยนต์สองขาให้มีการเดินที่มีเสถียรภาพและมั่นคงต่อสภาวะแวดล้อมที่ซับซ้อน เช่นทางชัน ขึ้นบันได ทางขรุขระ เป็นต้น และประหยัดพลังงานให้มากที่สุด โดยใช้กลวิธีการทางปัญญาประดิษฐ์ (Artificial Intelligence) โดยนำ Soft-computing ที่เป็นการรวมกันระหว่างหลักการของข่ายงานระบบประสาท (Neural Network) ระบบตรรกศาสตร์แบบคลุมเครือ (Fuzzy Logic System) และกลวิธีการถ่ายทอดพันธุกรรม (Genetic Algorithm) มาออกแบบระบบควบคุมลักษณะการเดิน (Gait generation) ของหุ่นยนต์สองขาทั้งแบบสถิตย์ (Static) และแบบพลวัติ (Dynamic)

1.3 แนวทางและแผนการดำเนินการวิจัย

- 3.1 ทำการศึกษาหลักการของข่ายงานระบบประสาท (Neural Network) ระบบตรรกศาสตร์แบบคลุมเครือ (Fuzzy Logic System) และกลวิธีการถ่ายทอดพันธุกรรม (Genetic Algorithm) และหลักในการรวมกันเป็น Soft-computing มาใช้ในการควบคุมหุ่นยนต์
- 3.2 เพื่อศึกษาการควบคุมหุ่นยนต์สองขา (Biped Walking Robot) โดยใช้แบบจำลองของ Double Inverted Pendulum (ศึกษาในลักษณะของ Computer Simulation) ในการจำลองการเดินร่วมกับการหลักการควบคุมของ นิวโรฟัชซี)

- 3.3 สร้างข้อมูลพื้นฐานในการหาสมดุลการทรงตัวของหุ่นยนต์สองขาในระหว่างเดินไปบนพื้นผิวราบ และพื้นผิวที่มีความชัน ให้กับหุ่นยนต์สองขาด้านแบบ Humanoid Robot (ซึ่งสร้างขึ้นที่ FIBO) โดยวิเคราะห์ลักษณะทางกายภาพของมนุษย์ในการเดิน วิ่งในสถานการณ์ต่างๆ โดยวัดการเคลื่อนไหวของมนุษย์จริง
- 3.4 ศึกษาและวิเคราะห์หาระบบควบคุมแบบ Soft-computing ในการควบคุมลักษณะการเดินของหุ่นยนต์ให้หุ่นยนต์สองขาให้มีการเดินที่มีเสถียรภาพและมั่นคงต่อสภาวะแวดล้อมที่ซับซ้อน เช่นทางชัน ขึ้นบันได ทางขรุขระ เป็นต้น
- 3.5 ทำการวิจัยทางด้าน Robot Vision โดยทำการวิจัยโดยใช้ทฤษฎีทางด้าน Image Processing เพื่อให้หุ่นยนต์สามารถคำนวณหาท่าทางและแนวการเดิน (Gait and Trajectory) เดินล่วงหน้า
- 3.6 การทดลอง
- 3.6.1 ทดลองการสมดุลย์ Inverted Pendulum โดยใช้ตัวควบคุมที่ได้ออกแบบไว้ (NEUROFUZZY Controller)
 - 3.1.2 ทดลองการทรงตัวและการเดินของหุ่นยนต์สองขา (Computer Simulation) โดยขยายผลการทดลองจากข้อ 6.3.1
 - 3.6.3 ทดลองการทรงตัวและการเดินของหุ่นยนต์สองขาในพื้นที่จริง โดยขยายผลการทดลองจากข้อ 6.3.2
- 3.7 สรุปผลการทดลอง
- 3.7.1 สรุปผลการทดลองที่ได้จากข้อ 6.
 - 3.7.2 จัดทำเอกสารเพื่อส่งตีพิมพ์ในวารสารทางวิชาการระหว่างประเทศ

ตารางเวลาขั้นตอนการดำเนินการวิจัย

ขั้นตอนการดำเนินงาน	ปีที่ 1 / เดือน												ปีที่ 2 / เดือน											
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12
1. ศึกษาหลักการของข่ายงานระบบประสาท (Neural Network) ระบบตรรกศาสตร์แบบคลุมเครือ (Fuzzy Logic System) และกลวิธีการถ่ายทอดพันธุกรรม (Genetic Algorithm) และหลักในการรวมกัน Soft-computing มาใช้ในการควบคุมหุ่นยนต์																								
2. ศึกษาการควบคุมหุ่นยนต์สองขา (Biped Walking Robot) โดยใช้แบบจำลองของ Double Inverted Pendulum (ศึกษาในลักษณะของ Computer Simulation) ในการจำลองร่วมกับการหลักการควบคุมของ นิวโรฟิซซี) และทำการทดลองกับอุปกรณ์จริง																								
3. สร้างข้อมูลพื้นฐานในการหาสมการการทรงตัวของหุ่นยนต์สองขา บนพื้นผิวราบ และพื้นผิวที่มีความชัน โดยวัดจากคนจริง																								
4 ศึกษาและวิเคราะห์หาระบบควบคุมแบบ Soft-computing ในการควบคุมลักษณะการเดินของหุ่นยนต์ให้หุ่นยนต์สองขาให้มีการเดินที่มีเสถียรภาพและมั่นคงต่อสภาวะแวดล้อมที่ซับซ้อน เช่นทางชัน ขึ้นบันได ทางขรุขระ และระบบ Vision ในการสร้างรูปแบบการเดินล่วงหน้า																								
5. รายงานความก้าวหน้าในรอบหกเดือนที่ผ่านมาและตีพิมพ์บทความ																								
6. ทดลองการทรงตัวและการเดินของหุ่นยนต์สองขา(Computer Simulation)																								
7. ทดลองการทรงตัวและการเดินของหุ่นยนต์สองขาในพื้นที่จริง โดยขยายผลจากการทดลองในข้อ 6																								
8. สรุปผลการวิจัยและเขียนรายงานการวิจัย สรุปทั้งหนึ่งปี																								
9 เขียนบทความลงวารสารต่างประเทศ																								
10 ทำการปรับปรุงแก้ไขระบบควบคุมการเดินสองขาและพัฒนา ระบบ vision เพื่อใช้หาเส้นทางเดินล่วงหน้า																								
11 เดินทางไปทำวิจัยที่ต่างประเทศและเขียนวิทยานิพนธ์																								
12. รายงานความก้าวหน้าในรอบหกเดือนที่ผ่านมา(ปีที่ 2)และตีพิมพ์บทความ																								
13. สรุปผลการวิจัยและเขียนรายงานการวิจัย สรุปทั้งหนึ่งปี(ปีที่ 2) และตีพิมพ์บทความ																								

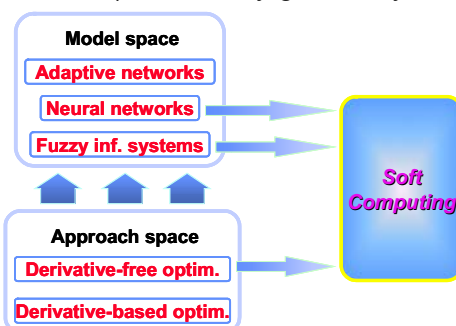
1.4 ผลที่คาดว่าจะได้รับ

- 4.1 พัฒนาศักยภาพทางด้านปัญญาประดิษฐ์ (Artificial Intelligence , AI)
- 4.2 สร้างข้อมูลในรูปของงานวิจัยพื้นฐานทางด้าน AI ที่เกี่ยวกับ Soft-computing (Neural Network, Fuzzy Logic, Genetic) เพื่อรองรับงานวิจัยในสายงานที่เกี่ยวข้องหรือสาขาอื่น
- 4.3 เป็นสื่อการเรียนการสอนในวิชาระบบควบคุมที่เกี่ยวกับการควบคุมความไม่สมดุลของระบบ
- 4.4 ปรับปรุงทฤษฎีการเรียนรู้ที่มีอยู่แล้วให้ดียิ่งขึ้นและองค์ความรู้ใหม่ให้เกิดขึ้น
- 4.5 เพื่อสร้างชื่อเสียงและเกียรติภูมิแก่ประเทศชาติ และสถาบันวิจัยหุ่นยนต์ภาคสนาม (FIBO) มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี(KMUTT)

บทที่ 2 ระบบการคำนวณเชิงอ่อน

2.1 บทนำ

การคำนวณเชิงอ่อน(Soft computing, SC) คือ ขบวนการในการออกแบบและพัฒนาระบบอัจฉริยะ เพื่อใช้ในการหาคำตอบของระบบที่เราสนใจภายใต้การอาศัยความสามารถของมนุษย์ในเรื่องของการใช้เหตุผลและการปรับตัวในสภาวะการที่ไม่แน่นอนและไม่ถูกต้อง การคำนวณเชิงอ่อนนี้ประกอบด้วยวิธีการหลายๆวิธี อาทิเช่น ข่ายงานระบบประสาท(Neural Network, NL) ตรรกศาสตร์คลุมเครือ(Fuzzy Logic, FL) การประมาณค่าเหตุผล(Approximate Reasoning, AR) การคำนวณเชิงสถิติ(Probabilistic Computing, PC) และวิธีการหาค่าเหมาะสม เช่นขบวนการเชิงพันธุกรรม(Genetic Algorithm, GA) หรือ Simulated Annealing ดังรูปที่ 2.1 ภายใต้ระบบการคำนวณเชิงอ่อนนี้ ตรรกศาสตร์คลุมเครือเป็นวิธีการที่เกี่ยวข้องกับค่าที่ไม่แน่นอนและการประมาณค่าเหตุผล ข้อมูลที่ใช้กับระบบนี้อยู่ในรูปตัวอักษรและพื้นฐานความรู้ของมนุษย์ ส่วนข่ายงานระบบประสาทเป็นระบบที่ใช้ในการหารูปแบบของระบบที่เราสนใจ(System Identification) โดยอาศัยพื้นฐานการเรียนรู้และการปรับเปลี่ยนของสมองมนุษย์ ส่วนกรรมวิธีเชิงพันธุกรรมนั้นเป็นกรรมวิธีการค้นข้อมูลหรือคำตอบแบบสุ่มอย่างมีระบบเพื่อหาจุดที่เหมาะสมสำหรับสิ่งที่เราสนใจ และการคำนวณเชิงสถิติหรือการประมาณค่าเหตุผลเป็นการวิเคราะห์การตัดสินใจภายใต้ความไม่แน่นอนของข้อมูล จะเห็นได้ว่าแก่นสารของการคำนวณเชิงอ่อนจะแตกต่างกับระบบแบบดั้งเดิม กล่าวคือเป็นระบบที่พิจารณาหาคำตอบภายใต้สิ่งแวดล้อมหรือสภาพที่เป็นจริง(Real World)[11] ดังนั้นหลักการของการคำนวณเชิงอ่อนคือการสร้างระบบที่อยู่ภายใต้ขอบเขตของความไม่แน่นอนและมีบางส่วนของจริงแน่นอนสัมผัสได้รวมถึงมีความคงทนต่อสภาวะที่เปลี่ยนแปลง (Robustness) ราคาถูก(low solution cost) และดีที่สุดสำหรับโลกแห่งความเป็นจริง(better within reality) ในความเป็นจริงถึงแม้ว่าข่ายงานระบบประสาท ตรรกศาสตร์แบบคลุมเครือ หรือ กรรมวิธีเชิงพันธุกรรม ต่างมีข้อดีและสามารถใช้กับระบบควบคุมในปัจจุบันก็ตาม แต่ก็ยังดีกว่าการประยุกต์หรือการนำมารวมกันของระบบทั้งสามแบบ ซึ่งระบบใหม่นี้มีชื่อเรียกว่า”ระบบอัจฉริยะผสม (Hybrid intelligent system)” ตัวอย่างเช่นการรวมกันของข่ายงานระบบประสาทและตรรกศาสตร์คลุมเครือ (Neuro-fuzzy system) หรือข่ายงานระบบประสาทและกรรมวิธีทางพันธุกรรม (Neuro-genetic system) หรือการรวมตัวของทั้งสามแบบ (Neuro-fuzzy-genetic system)



รูปที่ 2.1 แสดงองค์ประกอบของการคำนวณเชิงอ่อน [15]

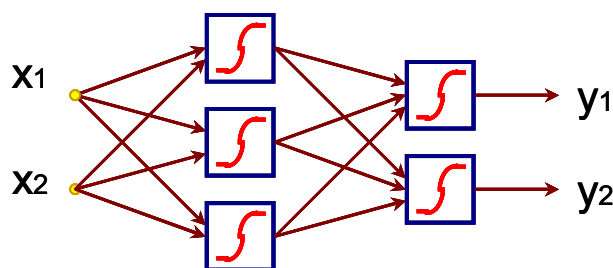
สำหรับในงานวิจัยนี้ได้นำเสนอถึงความพยายามที่จะนำระบบควบคุมแบบอัจฉริยะไปใช้ในการควบคุมหุ่นยนต์สองขาหรือหุ่นยนต์คล้ายคนเพื่อให้มีความสามารถคล้ายคลึงกับคนมากที่สุด เป็นที่ทราบกันดีแล้วหุ่นยนต์ได้เริ่มเข้ามามีบทบาทในชีวิตประจำวันของเรามากขึ้น จากทั้งในโรงงานอุตสาหกรรม การผลิตต่างๆ ไปจนถึงงานช่วยเหลือมนุษย์ในรูปแบบต่างๆ อาทิเช่นหุ่นยนต์ทำความสะอาดบ้าน หุ่นยนต์ดูแลผู้ป่วย เป็นต้น รูปแบบของหุ่นยนต์ก็เปลี่ยนแปลงไปในทางที่จะคล้ายคนมากที่สุด ดังเช่นในปี 1986 บ. ฮอนด้าได้เริ่มพัฒนาหุ่นยนต์ต้นแบบเรื่อยมาตั้งแต่ P1 P2 P3 [13] และ ASIMO กลุ่มนักวิจัยของ Kitano[14] ก็ได้ออกแบบและสร้างหุ่นยนต์คล้ายมนุษย์ชื่อ PINO และ Morph ขึ้นมาซึ่งเป็นหุ่นยนต์ขนาดเล็กและมีการติดต่อสื่อสารกับส่วนบังคับแบบไร้สายโดยใช้เทคโนโลยี Blue tooth เป็นต้น ยังมีความต้องการหุ่นยนต์ที่มีการทำงานคล้ายมนุษย์มากขึ้น หุ่นยนต์ควรมีระบบควบคุมที่มีความสามารถหลากหลาย รวมทั้งการรับข้อมูลข่าวสารจากสิ่งแวดล้อม และสามารถปรับเปลี่ยนตัวไปตามสิ่งแวดล้อม รวมถึงการที่ผู้ทำงานร่วมกับหุ่นยนต์หรือผู้ควบคุมหุ่นยนต์สามารถถ่ายทอดความรู้ความสามารถสู่หุ่นยนต์ เพื่อเพิ่มความสามารถให้กับหุ่นยนต์ได้ด้วย โดยเริ่มแรกเริ่มต้นจากการอาศัยติดตั้งระบบตรรกะต่างๆ ให้กับหุ่นยนต์เช่นระบบการมองเห็น ระบบวัดแรงกระทำที่ฝ่าเท้า ระบบการวัดมุมเอียงของลำตัว ระบบรับเสียง เป็นต้น แต่กระนั้นก็ตามหุ่นยนต์ก็ยังมีความสามารถในการปรับตัวให้เข้ากับสิ่งแวดล้อมที่เปลี่ยนแปลงไปน้อยอยู่ จึงจำเป็นต้องพัฒนาระบบควบคุมแบบอัจฉริยะเพื่อการนี้

ในงานวิจัยนี้เริ่มจากการนำเสนอระบบอัจฉริยะพื้นฐานในแต่ละแบบ หลักการทำงาน ข้อดี และข้อเสีย เพื่อนำไปสู่ระบบผสม(Hybrid Intelligent System) ซึ่งเป็นระบบที่อาศัยข้อดีของแต่ละระบบอัจฉริยะพื้นฐานมาผสมรวมกัน จากนั้นกล่าวถึงปัญหาของระบบควบคุมสำหรับหุ่นยนต์คล้ายคน จนถึงแนวทางออกแบบระบบควบคุมที่ใช้ในการควบคุมหุ่นยนต์เหล่านี้

2.2 องค์ประกอบของระบบควบคุมเชิงอ่อน (SC Components)

2.2.1 ข่ายงานระบบประสาท (Neural Network)

ข่ายงานระบบประสาท ได้ถูกเสนอครั้งแรกในปี 1943 โดย McCulloch และ Pitts โดยเสนอโครงข่ายที่มีความสามารถในการตัดสินใจระดับบิต จากนั้นได้มีการพัฒนาโครงข่ายระบบประสาทนี้เรื่อยมาในปัจจุบัน ข่ายงานระบบประสาทนี้เป็นรูปแบบการจำลองระบบประสาทของมนุษย์ขึ้นมา โดยอาศัยหลักการต่อกันของสมการเป็นร่างแหและมีการปรับค่าสัมประสิทธิ์ของการต่อกัน รูปแบบข่ายงานระบบประสาทที่นิยมคือแบบหลายชั้นป้อนไปข้างหน้า (Multi-layer feedforward networks) ดังรูปที่ 2.2 โครงสร้างแบบนี้อาศัยหลักการเรียนรู้แบบป้อนกลับ(Backpropagation learning) ในการปรับค่าตัวแปรในโครงข่ายโดยอาศัยข้อมูลเดิมของระบบ (Supervise learning) นอกจากนี้ก็ยังมีวิธีการเรียนรู้แบบไม่มีผู้ช่วยสอน โดยอาศัยเฉพาะรูปแบบข้อมูลขาเข้าเท่านั้น โดยสรุปข่ายงานระบบประสาทนั้นใช้ในเรื่องของการจำลองระบบที่เราสนใจ หรือสร้างระบบควบคุมโดยอาศัยชุดข้อมูลแบบเชิงตัวเลขในการปรับตัวและเรียนรู้



รูปที่ 2.2 แสดงแบบจำลองข่ายงานระบบประสาท [15]

ข้อดีของข่ายงานระบบประสาทคือมีความสามารถในการประมาณค่าฟังก์ชันหรือความสัมพันธ์ของระบบหรือคู่ข้อมูลเข้าและออกจากระบบใดระบบหนึ่ง แต่ข้อเสียคือ ความถูกต้องของระบบขึ้นกับความถูกต้องของข้อมูล และถึงแม้ว่าข้อมูลจะถูกต้องแต่ก็เป็นการประมาณค่าความสัมพันธ์เท่านั้น ไม่ใช่ความถูกต้องที่แท้จริง นอกจากนั้นยังใช้เวลาในการเรียนรู้กลุ่มข้อมูลนาน ต้องเรียนรู้ใหม่ถ้าข้อมูลใหม่อยู่นอกกลุ่มข้อมูลเดิม

2.2.2 ตรรกศาสตร์คลุมเครือ(Fuzzy Logic)

ในปี 1965 Zadeh ได้เสนอทฤษฎีตรรกศาสตร์คลุมเครือ (Fuzzy Sets Theory) ขึ้นทำให้เราสามารถใช้รูปแบบเชิงคุณภาพในการอธิบายคุณลักษณะระบบที่สนใจได้ ตรรกศาสตร์คลุมเครือใช้ตัวแปรเชิงบรรยายในการออกแบบจำลองระบบ ซึ่งตัวแปรเหล่านี้จะถูกแปลงจากปริมาณเชิงตัวเลข(Crisp Input) ให้อยู่ในรูปข้อมูลเชิงคลุมเครือ(Fuzzy Value) โดยใช้ฟังก์ชันสมาชิก(Membership function) เป็นตัวจัดการปริมาณเหล่านั้น จากนั้นจะถูกส่งไปยังขบวนการตีความซึ่งอาศัยกฎเงื่อนไข(If-Then Rule) ได้ค่าเชิงคลุมเครือออกมา นำไปผ่านขบวนการแปลงค่าให้เป็นเชิงตัวเลข ซึ่งเป็นค่าที่นำไปใช้งานต่อไป ขบวนการทางตรรกศาสตร์คลุมเครือเป็นดังรูปที่ 2.3 จะเห็นได้ว่าคล้ายๆกับข่ายงานระบบประสาทในเชิงการประมาณค่าความสัมพันธ์ แต่ไม่ได้อาศัยหลักการเรียนรู้ แต่เป็นการอาศัยหลักการการสร้างองค์ความรู้ของมนุษย์ ดังนั้นข้อมูลและรูปแบบจะเป็นในเชิงการบรรยายแทน



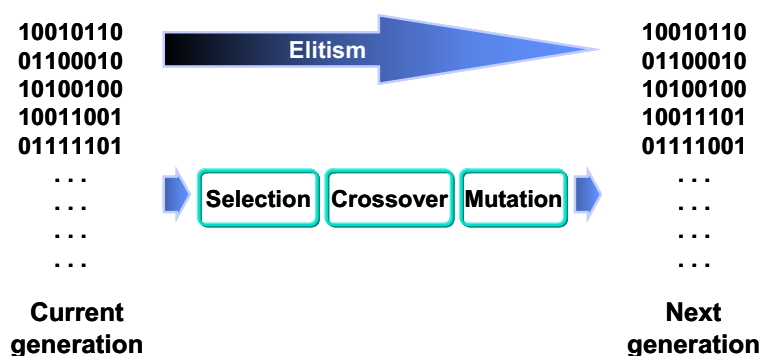
รูปที่ 2.3 แสดงแบบโครงสร้างของตรรกศาสตร์แบบคลุมเครือ

ดังนั้นด้วยกลไกทางด้านเหตุผลและคุณสมบัติในการประมาณค่าทำให้ตรรกศาสตร์คลุมเครือมั่นคง(Robustness) ในแต่ละตัวแปรต่างๆของระบบและต่อสิ่งรบกวน แต่กระนั้นก็ตามก็มีข้อเสียคือ ต้องใช้ผู้เชี่ยวชาญเฉพาะทางในการออกแบบระบบ ทำให้ต้องเสียเวลาในการออกแบบนาน

2.2.3 การคำนวณเชิงวิวัฒนาการ (Evolutionary Computing, EC)

การคำนวณเชิงวิวัฒนาการแสดงถึงพฤติกรรมในการปรับตัวที่ใช้กับระบบที่ไม่เป็นเชิงเส้น หลายมิติที่ปราศจากความต้องการความรู้พื้นฐานของปัญหานั้น ดังนั้นกลวิธีนี้มันคงต่อระบบที่พฤติกรรมเปลี่ยนแปลงไปตามเวลาแต่กระนั้นก็ตามการลู่เข้าสู่ค่าที่เหมาะสมก็ยังเป็นไปอย่างช้าอยู่ รูปแบบการคำนวณเชิงวิวัฒนาการนี้ มีด้วยกันหลายรูปแบบได้แก่ กลยุทธ์เชิงวิวัฒนาการ(Evolutionary Strategies, ES) โปรแกรมเชิงวิวัฒนาการ (Evolutionary Programming, EP) กลไกเชิงพันธุกรรม (Genetic Algorithms, Gas) และโปรแกรมเชิงพันธุกรรม (Genetic Programming, GP)

การคำนวณเชิงวิวัฒนาการพื้นฐานที่นำเสนอในบทความนี้เพื่อให้เข้าใจถึงกลไกการทำงานของวิธีการรูปแบบนี้ ได้แก่ กลไกเชิงพันธุกรรม ซึ่งเป็นวิธีการหาคำตอบที่เหมาะสม (Derivative free stochastic optimization) โดยใช้หลักการทางธรรมชาติและขบวนการวิวัฒนาการ ได้ถูกนำเสนอครั้งแรกในปี 1975 โดย John Michigan คุณสมบัติที่สำคัญของวิธีการนี้คือเป็นขบวนการค้นหาแบบขนาน (parallel search procedure) ใช้ได้กับทั้งปัญหาแบบต่อเนื่องและไม่ต่อเนื่อง



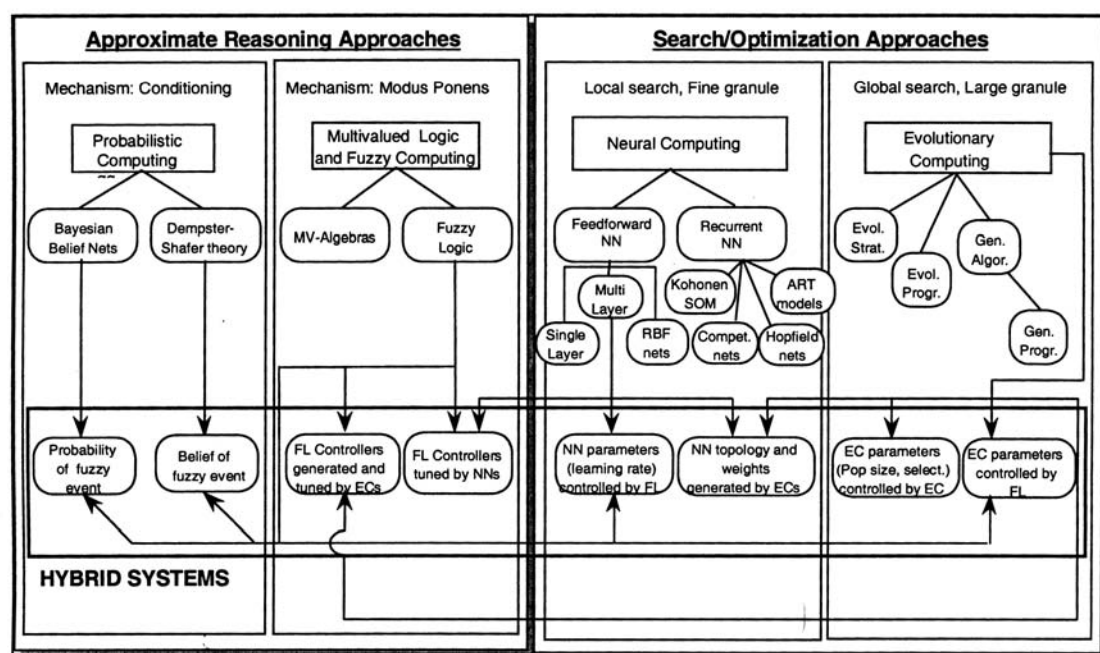
รูปที่ 2.4 แสดงขั้นตอนการสร้างประชากรรุ่นต่อไปของกลไกทางพันธุกรรม [15]

ในวิธีการเชิงพันธุกรรม ตัวแปรหรือคำตอบที่สนใจจะถูกทำให้อยู่ในรูปของกลุ่มเลขบิตฐานสอง ซึ่งเรียกว่า โครโมโซม(Chromosome) ซึ่งแต่ละตัวแปรจะมีค่าความฟิต (Fitness) อยู่ ในปัญหามากที่สุด เราจะสร้างสมการเป้าหมายขึ้นมาเพื่อใช้เป็นเกณฑ์ในการประเมินค่าความฟิตของแต่ละตัวแปร กลุ่มของข้อมูลเราเรียกว่า ประชากร (Population) ซึ่งในแต่ละครั้งของการประเมินหรือ generation นั้น จะเกิดกลุ่มประชากรใหม่ โดยเกิดจากการใช้ตัวเน้นการทางพันธุกรรมได้แก่ การข้าม (Crossover) และการกลายพันธุ์ (Mutation) ขบวนการนี้จะเกิดขึ้นซ้ำๆ จนกระทั่งได้ค่าที่ดีที่สุด ดังรูปที่ 2.4 ซึ่งเป็นการบรรยายขบวนการในการสร้างกลุ่มประชากรใหม่ (Next Generation) ขึ้นมา

2.2.4 การคำนวณเชิงอ่อน (Soft Computing, SC)

จากที่กล่าวมาข้างต้น ทำให้เราได้ทราบถึงองค์ประกอบต่างๆและความสามารถในการคำนวณเชิงอ่อน(SC) อันได้แก่ความสามารถในการเรียนรู้และการปรับตัว ความสามารถในการนำความรู้ของมนุษย์มาใช้งานในรูปแบบเชิงบรรยาย และความสามารถในการค้นหาคำตอบที่เหมาะสม ในการที่เราจะแก้ปัญหาในโลกแห่งความเป็นจริงนั้น เราพบว่าระบบมีความสลับซับซ้อนยากแก่การสร้างแบบจำลองระบบ และมีมากมายหลายคำตอบ ถ้าทำได้ก็ต้องใช้เวลาและสิ้นเปลืองค่าใช้จ่ายมาก วิธีการแก้ปัญหา

ด้วยการใช้การคำนวณเชิงอ่อนนี้ มีด้วยการสองแบบคือ ระบบที่ใช้ความรู้เป็นตัวขับเคลื่อน (Knowledge-driven) เช่น การคำนวณเชิงสถิติ และตรรกศาสตร์คลุมเครือ และ ระบบที่ใช้ข้อมูลเป็นตัวขับเคลื่อน (Data-Driven) เช่น ข่ายงานระบบประสาท และกลไกเชิงพันธุกรรม ดังรูปที่ 2.5



รูปที่ 2.5 แสดงแบบโครงสร้างของการคำนวณเชิงอ่อน(Soft computing technique)

และรูปแบบระบบอัจฉริยะผสม(Hybrid Intelligent System) [15]

ในระบบที่ใช้ความรู้เป็นตัวขับเคลื่อน (Knowledge-driven) นั้น กรรมวิธีในการออกแบบระบบจะใช้พื้นฐานในเชิงสัญญาณลักษณะ ปริมาณเชิงคุณภาพ หรือพื้นฐานความรู้จากองค์ความรู้ที่มีของระบบนั้นๆในการแสดงถึงพฤติกรรมของระบบ แต่ก็มีข้อจำกัดอยู่ที่พื้นฐานความรู้ที่เกี่ยวข้องกับระบบนั้นมีจำกัด ไม่สมบูรณ์ ไม่สามารถอธิบายระบบได้อย่างถูกต้อง โดยเฉพาะอย่างยิ่งระบบที่เปลี่ยนแปลงไปตามเวลา ส่วนในระบบที่ใช้ข้อมูลเป็นตัวขับเคลื่อน (Data-Driven) เป็นระบบที่ใช้วิธีการประมาณค่า (Regression) และเทคนิคการแบ่งกลุ่ม (Crisp Clustering) เพื่อหารูปแบบหรือพฤติกรรมของระบบที่เราสนใจจากข้อมูลที่ได้จากการทดลองหรือจากการสังเกตระบบ แต่ก็มีข้อจำกัดที่ความไม่สมบูรณ์ของข้อมูล และข้อมูลที่บิดเบือนไป(Noisy) หรือจากการเก็บข้อมูลที่ผิดพลาดจากอุปกรณ์วัด อุปกรณ์ตรวจวัดต่างๆ ซึ่งอาจสามารถปรับปรุงชุดข้อมูลด้วยวิธีการบางอย่างเช่น การจำแนกแยกแยะ(Feature Extraction) การกรอง (Filtering) หรือการประเมินข้อมูล (Data Validation)

ดังนั้นจะเห็นได้ว่าทั้งสองแบบของการคำนวณเชิงอ่อน(SC) นั้นก็ยังไม่สมบูรณ์ในตัวเอง จึงต้องมีการรวมทั้งสองแบบเข้าด้วยกันซึ่งเราเรียกว่า ระบบอัจฉริยะผสม (Hybrid Intelligent System) ซึ่งเป็นที่นิยมมากขึ้น เพราะว่าเราสามารถนำความรู้และเหตุผลเชิงบรรยายและข้อมูลเชิงตัวเลขที่ได้จากการสังเกตและทดลอง มาพัฒนาระบบและแก้ปัญหาที่สลับซับซ้อนได้ ตัวอย่างเช่น การรวมกันของข่ายงานระบบประสาทและตรรกศาสตร์คลุมเครือ (Neuro-fuzzy system) หรือข่ายงานระบบประสาทและกรรมวิธีทางพันธุกรรม (Neuro-genetic system) หรือการรวมตัวของทั้งสามแบบ (Neuro-fuzzy-genetic)

system) แต่สำหรับในบทความนี้จะเน้นที่ระบบรวมระหว่างการรวมกันของข่ายงานระบบประสาทและตรรกศาสตร์คลุมเครือ (Neuro-fuzzy system) อันมีหลายชนิด และจะได้กล่าวถึงชนิดต่างๆ รวมถึงข้อดี ข้อเสียของแต่ละแบบในบทถัดไป

2.3. ระบบอัจฉริยะผสม (Hybrid Intelligent System)

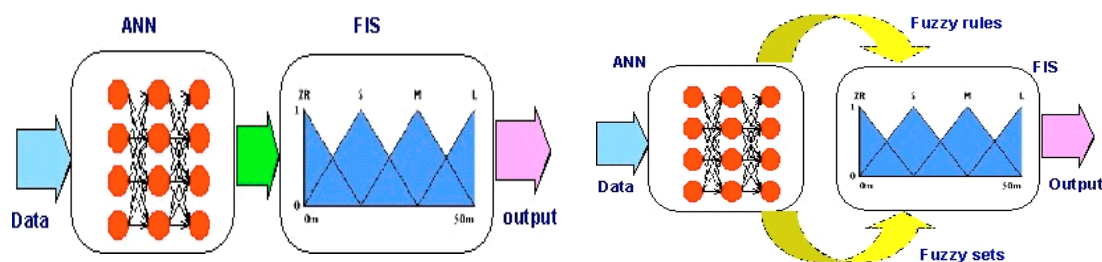
2.3.1 ระบบผสมระหว่างข่ายงานระบบประสาทและตรรกศาสตร์คลุมเครือ (Neuro-Fuzzy System)

เป็นที่ทราบกันดีแล้วว่าตรรกศาสตร์แบบคลุมเครือและข่ายงานระบบประสาทต่างก็มีข้อดี และข้อเสียภายในตัวของมันเอง และได้ถูกประยุกต์ใช้กับการควบคุมระบบต่างๆ รวมทั้งในเรื่องของระบบการมอง การตัดสินใจ มาเป็นเวลาหลายสิบปี ซึ่งเราสามารถสรุปคุณลักษณะของทั้งสองแบบได้ดังรูปที่ 2.6 จากรูปจะเห็นได้ว่า ทั้งสองแบบยังไม่สมบูรณ์ในตัวเอง แต่ถ้านำระบบทั้งสองมารวมกันจะทำให้ได้ระบบที่สมบูรณ์ยิ่งขึ้น เราเรียกระบบใหม่นี้ว่า **“นิวโรฟัซซี่ (Neuro-fuzzy system)”**

Neural Networks	Fuzzy Inference System
Prior rule-based knowledge cannot be used	Prior rule-based can be incorporated
Learning from scratch	Cannot learn (use linguistic knowledge)
Black box	Interpretable (if-then rules)
Complicated learning algorithms	Simple interpretation and implementation
Difficult to extract knowledge	Knowledge must be available

รูปที่ 2.6 แสดงตารางการเปรียบเทียบการใช้งานและคุณลักษณะ
ระหว่างข่ายงานระบบประสาทและตรรกศาสตร์แบบคลุมเครือ [15]

ในปัจจุบันระบบรวมนิวโรฟัซซี่เป็นที่นิยมมากขึ้นในการแก้ปัญหาที่สลับซับซ้อน เนื่องจากถ้าเรามีองค์ความรู้(Knowledge) ของระบบนั้น เราสามารถเริ่มจากสร้างกฎเชิงบรรยายของตรรกศาสตร์คลุมเครือ หรือ ข้อมูลเชิงตัวเลขของระบบเราก็สามารถเริ่มจากการสร้างข่ายงานระบบประสาทและใช้กระบวนการเรียนรู้ของข่ายงานระบบประสาทได้ ในการออกแบบระบบรวมนิวโรฟัซซี่นั้น รูปแบบเริ่มแรกก็คือการนำมำทำงานร่วมกัน(Concurrent model) มีด้วยกันสองรูปแบบคือนำข้อมูลมาผ่านข่ายงานระบบประสาทก่อนแล้วค่อยนำข้อมูลไปประเมินผลด้วยตรรกศาสตร์แบบคลุมเครือ เรียกว่า **“Concurrent NF System”** อีกแบบคือการนำข้อมูลมาผ่านตรรกศาสตร์แบบคลุมเครือก่อนแล้วค่อยนำข้อมูลไปประเมินผลด้วยตรรกศาสตร์แบบคลุมเครือ รูปแบบนี้ เรียกว่า **“Concurrent FN System”** ดังรูปที่ 2.7 ก. แสดงรูปแบบผสมแบบ **“Concurrent NF System”**



รูปที่ 2.7 ก. แสดงรูปแบบผสมนิวโรฟัซซี่แบบ Cooperative

ข. แสดงรูปแบบผสมนิวโรฟัซซี่แบบ Cooperative model [15]

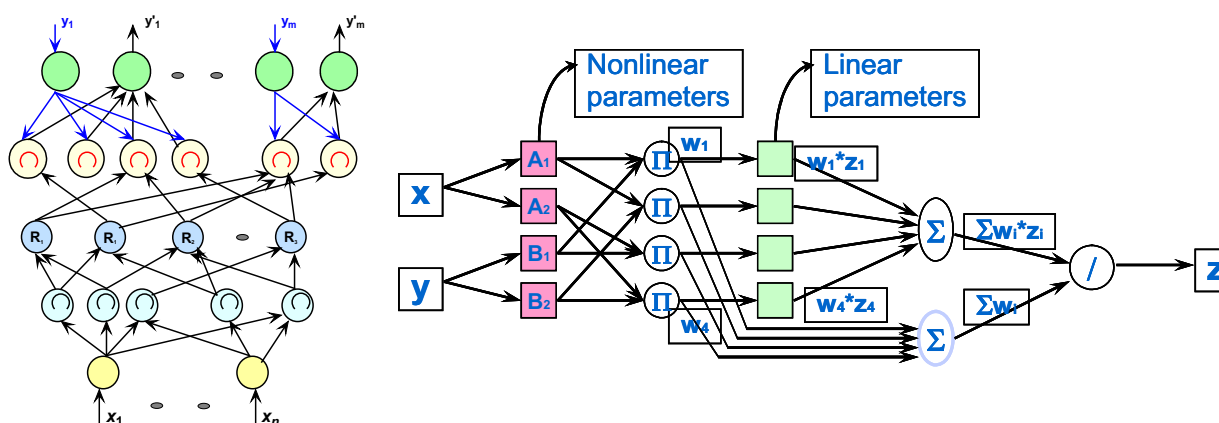
ส่วนอีกแบบเป็นการนำข่ายงานระบบประสาทไปหาตัวแปรที่สำคัญของตรรกศาสตร์แบบคลุมเครือ เราเรียกวิธีการนี้ว่า **“Cooperative model”** เนื่องจากถึงแม้ว่าตรรกศาสตร์คลุมเครือนั้นมีความสามารถในการนำความรู้ของมนุษย์มาใช้ในรูปของกฎ แต่มันก็ใช้เวลาในการสร้างกฎและต้องอาศัยผู้ที่มีความรู้จริง ส่วนข่ายงานระบบประสาทนั้นมีความสามารถในการปรับตัว จึงเป็นวิธีการที่ดีที่จะนำกลวิธีการเรียนรู้ของข่ายงานระบบประสาทมาใช้ในการปรับค่าสมการสมาชิก (Membership function) และกฎของตรรกศาสตร์คลุมเครือ ดังรูปที่ 2.7 ข.

3.2 รูปแบบการรวมกันของระบบนิวโรฟัซซี่ (Fused Neuro-Fuzzy System)

โดยทั่วไปการนำกลวิธีการเรียนรู้มาประยุกต์ใช้กับตรรกศาสตร์คลุมเครือนั้น ต้องมีการปรับโครงสร้างในรูปข่ายงานระบบประสาท เนื่องจากกลวิธีการเรียนรู้แบบค่อยๆ ลดลง (Gradient descent) นั้นใช้ได้กับฟังก์ชันต่อเนื่อง ในขณะที่ช่วงการตีความ (Inference System) ของตรรกศาสตร์คลุมเครือเป็นฟังก์ชันไม่ต่อเนื่อง ดังนั้นวิธีการโดยทั่วไปจึงเป็นการสร้างตรรกศาสตร์คลุมเครือที่มีโครงสร้างคล้ายคลึงกับข่ายงานระบบประสาท และมีการปรับกระบวนการตีความให้เป็นฟังก์ชันต่อเนื่องและใช้กระบวนการเรียนรู้ที่เหมาะสม รูปแบบการนำตรรกศาสตร์แบบคลุมเครือมารวมกับข่ายงานระบบประสาทนั้น มีผู้เสนอไว้ต่าง ๆ ดังนี้ FALCON[16], ANFIS[15], GARIC[17], NEFCON[18], FUN[19], SONFIN[20], EFUNN[21], DMEFNN[22], Evaluation design of Neuro-Fuzzy [23]

- **Fuzzy Adaptive learning control network (FALCON)**

ประกอบด้วย 5 ชั้นดังรูปที่ 2.8 ก. และมีสองโหนดเชิงบรรยายในแต่ละตัวแปรเอาต์พุต โหนดหนึ่งสำหรับเอาต์พุตที่ใช้สอน (Training Output) และ อีกโหนดสำหรับเอาต์พุตที่ได้จากข่ายงาน (Actual Output) ในชั้นที่หนึ่งของชั้นซ่อนนั้นเป็นการแปลงค่าเชิงตัวเลขเป็นค่าเชิงตรรกะ (Fuzzification) โดยแต่ละโหนดแทนแต่ละฟังก์ชันสมาชิก ชั้นซ่อนที่สอง สามต่อไปเป็นชั้นที่เกี่ยวข้องกับกฎของฟัซซี่ ส่วนชั้นซ่อนสุดท้ายเป็นชั้นแปลงค่าทางตรรกะเป็นค่าเชิงตัวเลข (Defuzzification) FALCON ใช้การเรียนรู้แบบผสมในการสอน กล่าวคือใช้การสอนแบบไม่มีผู้ช่วยสอน (Unsupervised Learning) ในการกำหนดค่าฟังก์ชันสมาชิกและกฎฟัซซี่ตอนเริ่มต้น และใช้วิธีการเรียนรู้แบบค่อยๆ ลด (Gradient decent learning) ในการปรับค่าฟังก์ชันสมาชิกเพื่อให้ได้เอาต์พุตตามต้องการ



รูปที่ 2.8 ก. แสดงข่ายงานระบบประสาทฟัซซี่แบบ FALCON

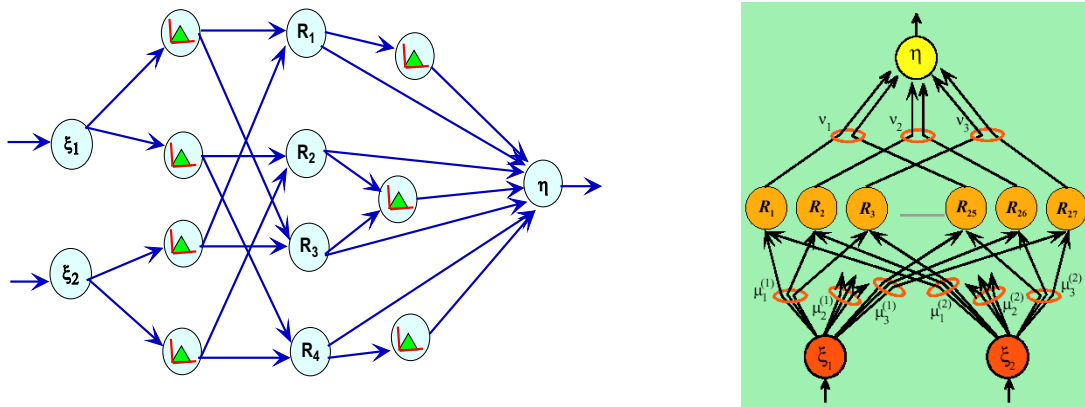
ข. แสดงข่ายงานระบบประสาทฟัซซี่แบบ ANFIS

● Adaptive Neuro Fuzzy Inference System (ANFIS)

ANFIS ได้ถูกเสนอขึ้นโดย Jang R ในปี 1992 เป็นระบบตรรกศาสตร์แบบคลุมเครือแบบ Takagi Sugeno ประกอบด้วย 5 ชั้นด้วยกัน ดังรูปที่ 2.8 ข. ในชั้นช่อนที่ 1 นั้นเป็นส่วนที่ใช้ในการแปลงค่าเชิงตัวเลขไปสู่ค่าเชิงตรรกะ และในชั้นที่สองเป็นส่วนที่ใช้ประเมินค่ากฎของฟัซซี่โดยใช้ตัวปฏิบัติการแบบ T-norm ในชั้นที่สามเป็นส่วนที่ใช้ แปลงค่าที่ได้จากกฎให้อยู่ในระดับเดียวกัน และส่งไปยังชั้นช่อนที่ 4 จะคำนวณหาค่าตัวแปรของส่วนเอาท์พุต และชั้นสุดท้ายก็จะเป็นชั้นที่รวมผลของแต่ละเอาท์พุตที่ได้จากแต่ละกฎ ANFIS ใช้การเรียนรู้แบบแพร่กระจายกลับ (Backpropagation Learning) ในการปรับค่าตัวแปรของฟังก์ชันสมาชิก และใช้วิธีการค่าเฉลี่ยกำลังสองน้อยสุด (least Mean Square) ในการค่าตัวแปรของสมการเอาท์พุต การเรียนรู้และการปรับค่าถือเป็นแบบมีผู้ช่วยสอน

● Generalized Approximate Reasoning based Intelligent Control (GARIC)

GARIC เป็นรูปแบบของตัวควบคุมแบบนิวโรฟัซซี่ โดยประกอบด้วยข่ายงานระบบประสาทสองชุด คือ ส่วนที่ใช้เลือกพฤติกรรม (Action Selection Network, ASN) และส่วนที่ใช้ประเมินค่าสถานะ (Action State Evaluation, AEN) ซึ่งส่วนนี้เป็นส่วนที่ใช้ประเมิน ASN ใน ASN นั้น เป็นข่ายงานระบบประสาทแบบห้าชั้น ดังในรูปที่ 2.9 ก. แต่การเชื่อมต่อแต่ละชั้นจะไม่มีค่าถ่วงน้ำหนักมาเกี่ยวข้อง ในชั้นช่อนที่แรกจะเก็บค่าตัวแปรเชิงบรรยายของแต่ละข้อมูลเข้าไว้ ในชั้นช่อนที่สองเป็นส่วนที่เกี่ยวข้องกับกฎของฟัซซี่ และประเมินผลโดยใช้ตัวปฏิบัติการ Softmin ส่วนชั้นช่อนที่สามเป็นส่วนที่ใช้ประเมินค่าเชิงบรรยายของข้อมูลขาออก GARIC ใช้วิธีการค่าเฉลี่ยของค่าสูงสุด (Local mean of Maximum) ในการคำนวณหาผลที่ได้จากกฎ เพื่อหาค่าเอาท์พุต และในการเรียนรู้ใช้วิธีการผสมระหว่างวิธีการลดค่าลง (Gradient descent) และ การเรียนรู้เชิงบังคับ (Reinforcement learning) ในการปรับค่าพารามิเตอร์ต่างๆในระบบควบคุม



รูปที่ 2.9 ก. แสดงข่ายงานระบบประสาทฟัซซี่แบบ GARIC ในส่วนของ ASN

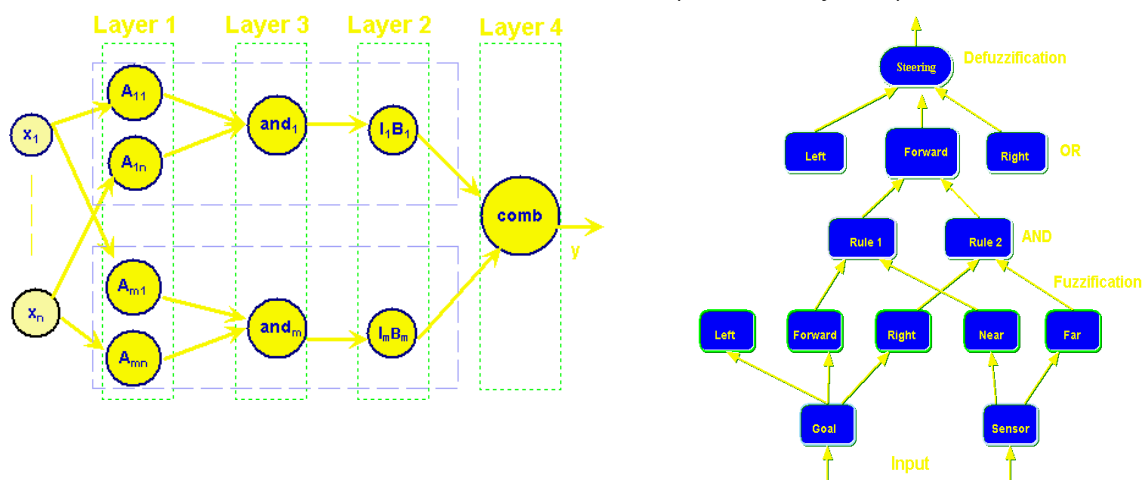
ข. แสดงข่ายงานระบบประสาทฟัซซี่แบบ NEFCON

- **Neuro-Fuzzy Control : NEFCON**

NEFCON เป็นรูปแบบการตีความแบบ Mandani ดังรูปที่ 2.9 ข. การเชื่อมต่อระหว่างชั้นในโครงข่าย เชื่อมโยงโดยใช้ค่าฟัซซี่และกฎของฟัซซี่ แทนค่าถ่วงน้ำหนัก ขบวนการเรียนรู้ของระบบนี้เป็นแบบผสมระหว่างการเรียนรู้เชิงบังคับ (Reinforcement learning) และการเรียนรู้แบบแพร่กระจายกลับ (Backpropagation) NEFCON สามารถเรียนรู้และปรับกฎของฟัซซี่ที่เหมาะสมที่สุดได้ นอกจากนี้ยังแบ่งเป็นสองรูปแบบตามการใช้งานคือ สำหรับการประมาณค่าฟังก์ชัน (Function Approximation) และ สำหรับปัญหาในการแบ่งกลุ่ม (Classification)

- **Fuzzy Inference and Neural Network in fuzzy inference software (FINEST)**

FINEST มีความสามารถในการปรับเปลี่ยนสองอย่างคือ การปรับเปลี่ยนค่าตัวทำนายฟัซซี่ (fuzzy predicates) และ ฟังก์ชันเป็นนัย (implication function) FINEST ใช้การเรียนรู้แบบแพร่กระจายกลับ ในการปรับค่าพารามิเตอร์ต่างๆ ดังในรูปที่ 2.10 ก. ในรูปนี้แสดงให้เห็นถึงโครงสร้างในแต่ละชั้น และขบวนการคำนวณของการประเมินค่าทางฟัซซี่ (Inference system)



รูปที่ 2.10 ก. แสดงข่ายงานระบบประสาทฟัซซี่แบบ FINEST

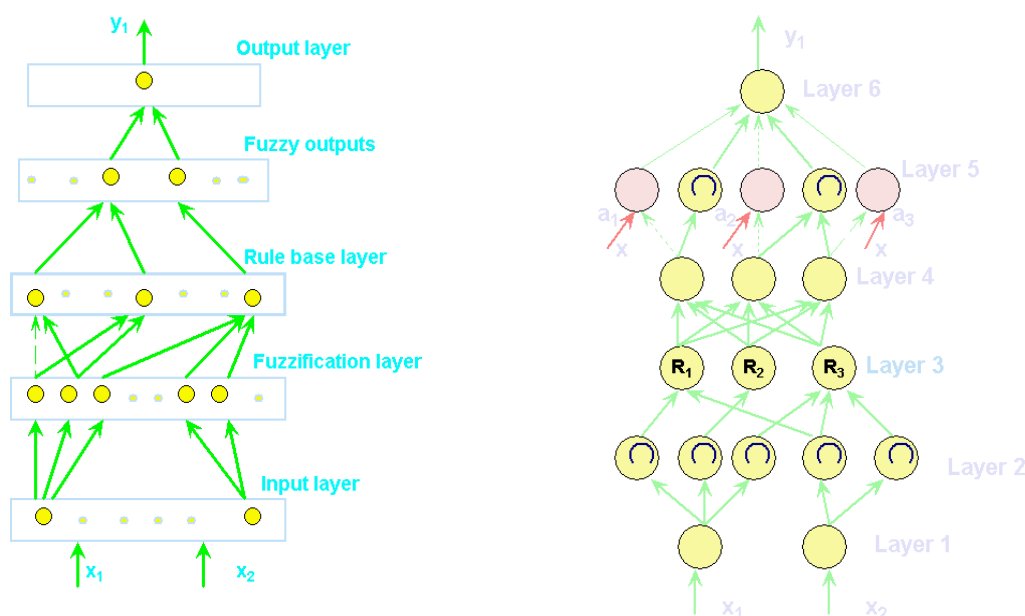
ข. แสดงข่ายงานระบบประสาทฟัซซี่แบบ FUN

- **Fuzzy Net (FUN)**

ใน FUN ดังรูปที่ 2.10 ข. นั้น ชั้นซ่อนที่ 1 เป็นชั้นที่เป็นค่าของฟังก์ชันสมาชิกของแต่ละข้อมูลขาเข้า ในชั้นซ่อนที่สองเป็นส่วนที่เกี่ยวข้องกับกฎทางฟัซซี่โดยเชื่อมกับชั้นที่หนึ่งด้วยฟัซซี่แอนด์ (Fuzzy-AND) ในชั้นซ่อนที่3 เป็นชั้นที่เก็บค่าของฟังก์ชันสมาชิกของข้อมูลขาออก และมีฟังก์ชันกระตุ้นเป็นฟัซซี่ออล (Fuzzy-OR) ส่วนในชั้นสุดท้ายเป็นชั้นข้อมูลขาออกที่ใช้กระบวนการแปลงค่าฟัซซี่เป็นค่าเชิงปริมาณ (Defuzzification) กระบวนการเรียนรู้ของ FUN นั้นใช้กระบวนการเชิงสถิติ (Stochastic process) ในการหาค่าฟังก์ชันสมาชิก โดยใช้ฟังก์ชันราคา (Cost function) เป็นเกณฑ์ในการประเมิน

- **Evolving Fuzzy Neural Network (EFuNN)**

ใน EFuNN ดังรูปที่ 2.11 ก. นั้น โหนดของโครงข่ายจะถูกสร้างระหว่างกระบวนการเรียนรู้ ชั้นข้อมูลขาเข้าจะส่งข้อมูลไปสู่ชั้นที่สอง ซึ่งเป็นชั้นที่ใช้คำนวณหาค่าฟังก์ชันสมาชิกของแต่ละตัวแปรข้อมูลขาเข้า ส่วนในชั้นที่สามเป็นชั้นที่เกี่ยวข้องกับกฎของฟัซซี่ (If-then Rule) ซึ่งเกิดจากการประเมินชุดข้อมูลขาเข้าและขาออก โดยแต่ละโหนดของกฎจะมีสองเวกเตอร์ค่าถ่วงน้ำหนัก ซึ่งจะถูกรับอีกครั้งในกระบวนการเรียนรู้ ต่อมาในชั้นที่สี่เป็นชั้นของค่าฟังก์ชันสมาชิกของข้อมูลขาออก และชั้นที่ห้าเป็นชั้นที่แปลงค่าข้อมูลฟัซซี่เป็นข้อมูลเชิงปริมาณ นอกจากนี้ยังมี Dynamic Evolving Fuzzy Neural network (dmEFuNN) ซึ่งเป็นรูปแบบที่พัฒนาต่อจาก EFuNN ซึ่งแตกต่างกันตรงที่ EFuNN ใช้กระบวนการตีความฟัซซี่แบบ Mamdani ส่วน dmEFuNN ใช้กระบวนการตีความฟัซซี่แบบ Takagi-Sugeno



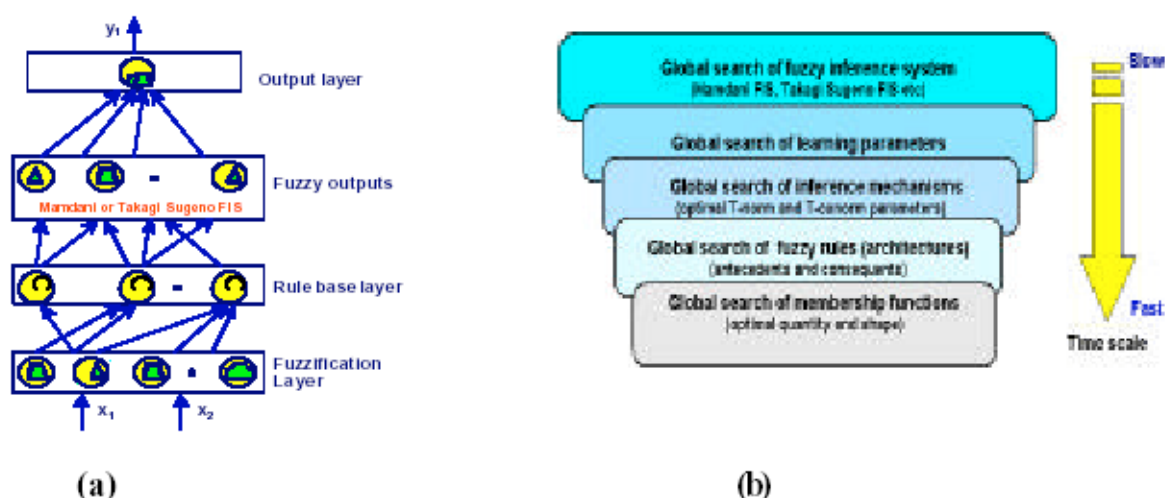
รูปที่ 2.11 ก. แสดงข่ายงานระบบประสาทฟัซซี่แบบ EFuNN

ข. แสดงข่ายงานระบบประสาทฟัซซี่แบบ SONFIN

● Self constructing Neural Fuzzy Inference Network (SONFIN)

SONFIN เป็นโครงข่ายที่ใช้รูปแบบการตีความฟัซซี่แบบ Takagi-Sugeno มีลักษณะดังรูปที่ 2.11 ข. จากข้อมูลขาเข้าที่มี นำมาพิจารณาและสร้างชั้นของฟังก์ชันสมาชิก ด้วยกระบวนการแบ่งกลุ่ม (Clustering Process) รวมทั้งกฎที่ต้องเกี่ยวข้องกับฟังก์ชันสมาชิกในชั้นถัดไป และในชั้นถัดไปที่ใช้หาค่าข้อมูลขาออกที่เกี่ยวข้องกับสมการเชิงเส้นซึ่งประกอบด้วยค่าข้อมูลค่าเข้าและค่าตัวแปรสัมประสิทธิ์ต่าง ๆ นั้น จะถูกปรับเปลี่ยนในกระบวนการเรียนรู้ โดยใช้วิธีค่าเฉลี่ยกำลังสองน้อยสุด (Least Mean Square) หรือ ขบวนการค่าเฉลี่ยกำลังสองแบบย้อน (Recursive Mean Square) และตัวแปรในส่วนข้อมูลขาเข้านั้นปรับค่าโดยกระบวนการแพร่กระจายย้อนกลับ (Back propagation)

● Evolutionary Design of Neuro-Fuzzy System



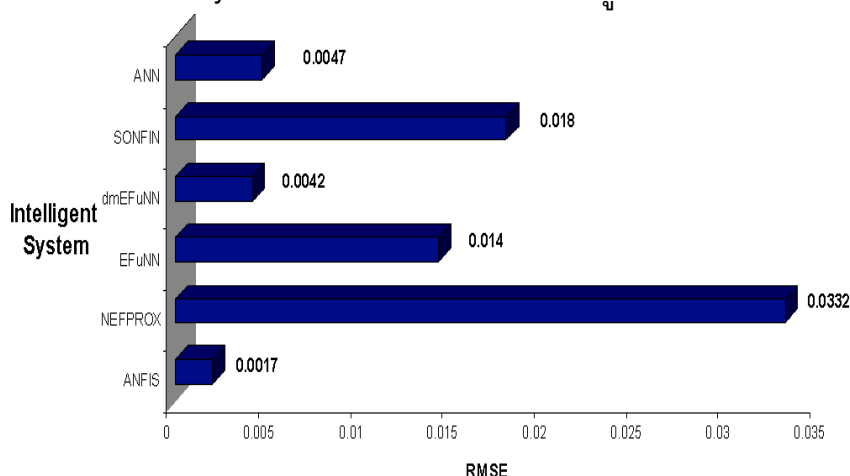
รูปที่ 2.12 แสดงขบวนการระบบประสาทฟัซซี่แบบ Evolutionary Design of Neuro-Fuzzy System

ในขั้นตอนการออกแบบเชิงวิวัฒนาการของระบบนิวโรฟัซซี่ชนิดนี้ แต่ละโหนด รูปแบบโครงสร้างและกระบวนการเรียนรู้ในการปรับค่าพารามิเตอร์ต่างๆ จะอยู่บนพื้นฐานของกระบวนการค้นหาเชิงวิวัฒนาการห้าขั้นตอน ดังรูปที่ 2.12 ด้านขวา ซึ่งขบวนการนี้สามารถใช้ได้กับทั้งรูปแบบการตีความแบบ Mamdani และ แบบ Takagi-Sugeno รูปแบบโครงสร้างของระบบนิวโรฟัซซี่ชนิดนี้เป็นดังรูปที่ 2.12 ซ้ายมือ กล่าวคือใช้กระบวนการสุ่มเชิงวิวัฒนาการค่าที่เหมาะสมของ ชนิดของการตีความ จำนวนโหนดในแต่ละชั้นและรูปแบบการเชื่อมโยงโครงข่าย นอกจากนี้ในรูปที่ 2.12 ขวามือแสดงให้เห็นถึงการใช้ขบวนการค้นหาโดยรวม (Global Search Process) ในการหาค่าพารามิเตอร์ต่างๆ เช่น ตัวแปรในการเรียนรู้กลไกของการตีความ กฎต่างๆ และฟังก์ชันสมาชิก เป็นต้น สิ่งสำคัญที่สุดสำหรับระบบนี้ก็คือความรู้เบื้องต้น ยิ่งมีความรู้มากเท่าไรก็จะได้โครงข่ายระบบที่มีคุณภาพดีที่สุดด้วย

จากโมเดลของNeuro-Fuzzy ที่กล่าวมาทั้งหมด สามารถสรุปรายละเอียดโดยรวมได้ดังนี้

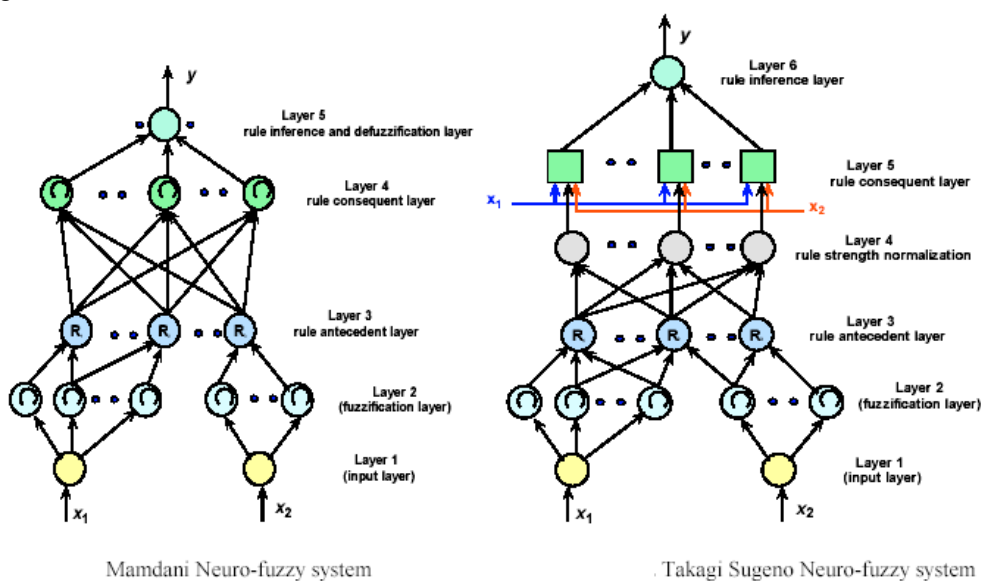
1. Neuro-Fuzzy model มีทั้งความสามารถในการตีความและการเรียนรู้ในรูปแบบมีผู้สอน (Supervised model)

2. GARIC, FALCON, ANFIS, NEFCON, FUN, SONFIN เป็นแบบที่การปรับค่า พารามิเตอร์ของระบบภายใต้โครงสร้างระบบที่แน่นอน
3. EFUNN, DMEFUNN ใช้การเรียนรู้เพียงหนึ่งครั้ง ทำให้เหมาะสำหรับการเรียนรู้แบบทันที (Online Learning)
4. Evaluation design of Neuro-Fuzzy เป็นแบบที่ใช้หลักการประเมินค่าเข้ามาช่วย ทำให้สามารถนำไปใช้กับการแก้ไขปัญหาที่หาค่าที่เหมาะสม
5. การเปรียบเทียบประสิทธิภาพของการเรียนรู้ของ Neuro-Fuzzy model แต่ละแบบ โดยทดลองใช้กับการทำนาย Mackey-Glass chaotic time series ดังรูปที่ 2.13



รูปที่ 2.13 แสดงค่าความผิดพลาดกำลังสองน้อยสุด (RMSE) ของระบบนิวโรฟัซซีแต่ละแบบ

6. ถ้าพิจารณาในเทอมของ RMSE error พบว่า Neuro-Fuzzy model ที่ใช้โครงสร้างของ Takagi Sugeno FIS มีประสิทธิภาพ ดีกว่า Mamdani FIS



รูปที่ 2.14 แสดงข่ายงานระบบประสาทฟัซซี แบบ Mamdani FIS และแบบTakagi Sugeno FIS

7. ในทอมการเรียนรู้ Neuro-Fuzzy model ใช้รูปแบบการเรียนรู้แบบ gradient descent techniques และ backpropagation ซึ่งทำให้การเรียนรู้เป็นไปอย่างช้า ควรหาวิธีการเรียนรู้ที่เร็วขึ้นอาทิเช่นแบบ conjugate gradient search เป็นต้น

2.4. การนำไปประยุกต์ใช้กับหุ่นยนต์สองขา (Application to biped robots)

2.4.1 ปัญหาในการควบคุมหุ่นยนต์สองขา (Control Problems in Humanoid robot)

ในการพัฒนาและวิจัยของหุ่นยนต์คล้ายมนุษย์หรือหุ่นยนต์สองขานั้น โดยส่วนใหญ่เน้นหนักไปที่กลไกและส่วนประกอบของหุ่นยนต์ อันได้แก่ระบบขับเคลื่อน(Actuators) ระบบตรวจวัดต่างๆ (Sensors) วัสดุที่ใช้ผลิตชิ้นส่วน รวมถึงระบบควบคุมอิเล็กทรอนิกส์ คอมพิวเตอร์ เป็นต้น เพื่อมุ่งหวังให้หุ่นยนต์สามารถทำงานได้หลากหลาย ในระบบควบคุมการเดินของหุ่นยนต์นั้น ปัญหาหลักที่สำคัญคือ หุ่นยนต์มีความไม่เป็นเชิงเส้นสูง อันเนื่องมาจากมีจำนวนมอเตอร์มากและกลไกการเคลื่อนที่ที่ซับซ้อน นอกจากนี้ยังมีการเปลี่ยนแปลงที่ไม่ต่อเนื่องในขณะที่ทำการเดินอีกด้วย ดังนั้นระบบควบคุมจำเป็นต้อง มีเสถียรภาพสูง รวดเร็ว และมีความสามารถในการวิเคราะห์ลักษณะการเดินแบบต่างๆ

การเดินของหุ่นยนต์สามารถแบ่งได้เป็นสามแบบคือ การเดินแบบสถิตยศาสตร์ (Static Walking) เป็นการเดินแบบช้าๆ โดยพยายามรักษาจุดรวมศูนย์กลางมวลให้อยู่ภายใต้พื้นที่รองรับเช่นที่ฝ่าเท้าข้างใดข้างหนึ่งเมื่อยืนอยู่ขาเดียว หรือระหว่างขาทั้งสองข้างเมื่อยืนด้วยขาทั้งสองพร้อมกัน แบบที่สองคือการเดินแบบกึ่งพลวัต (Dynamic Walking) เป็นการเดินแบบที่เร็วและขาทั้งสองข้างมีช่วงเวลาที่ขาทั้งสองสัมผัสพื้นพร้อมกันน้อยมาก และหุ่นยนต์แบบสามารถเดินช้าๆแบบสถิตยศาสตร์ได้ด้วย ส่วนแบบสุดท้ายเป็นแบบพลวัตบริสุทธิ์ (Purely Dynamic) หุ่นยนต์มีลักษณะที่ปราศจากฝ่าเท้า แต่สามารถเดินแบบพลวัตได้ แต่ไม่สามารถเดินแบบสถิตยศาสตร์ได้ เราพบว่าปัญหาในการควบคุมการเดินแบบพลวัตนั้นมีความยุ่งยากและซับซ้อนกว่าการเดินแบบสถิตยศาสตร์มาก จึงได้มีผู้คิดค้นกฎเกณฑ์เพื่อรักษาสสมดุลขึ้นหลายๆกฎ แต่ที่นิยมมากที่สุดคือ **จุดรวมโมเมนต์เป็นศูนย์ (Zero Moment Point, ZMP)**

นักวิจัยด้านกลไกทางชีวของมนุษย์ ได้ทำการศึกษาท่าทางการเดินของมนุษย์ และได้สรุปว่ารูปแบบการเดินมนุษย์ที่เป็นจังหวะนั้น เป็นการรวมไว้สองแบบคือ การเดินที่มีจังหวะที่แน่นอนตามธรรมชาติ(Inherent pattern) และแบบกิริยาตอบสนอง (Reflexive Action) การเดินแบบแรกนั้นจะเป็นจังหวะตามคาบเวลา โดยพิจารณาจากรูปแบบการเดินที่เหมาะสมที่สุดภายใต้การเดินที่ไม่มีการรบกวน ส่วนแบบที่สองเป็นการตอบสนองอย่างทันท่วงทีโดยอาศัยข้อมูลจากระบบการตรวจวัดต่างๆ การปรับเปลี่ยนท่าทางนี้อาศัยเกณฑ์ทางเสถียรภาพของการเดินมาพิจารณา ตัวอย่างเหตุการณ์ได้แก่ แหล่งรบกวนจากระบบและเหตุการณ์ที่ไม่คาดคิด ในกรณีนี้ระบบควบคุมต้องมีความอัจฉริยะในการปรับค่าต่างๆ เพื่อรักษาการเดินอย่างมีประสิทธิภาพให้ได้

2.4.2 การใช้ข่ายงานระบบประสาทในการควบคุมหุ่นยนต์สองขา

ในอดีตมีนักวิจัยจำนวนมากที่ใช้ข่ายงานระบบประสาทในการควบคุมหุ่นยนต์สองขา นิยมใช้ข่ายงานระบบประสาทในเรื่องการสังเคราะห์ท่าทางการเดินและการควบคุมหุ่นยนต์สองขา ข่ายงานที่นิยมใช้ได้แก่ ข่ายงานหลายชั้น(Multilayer perception) CMAC Network fuzzy-neural network RBF

network Hopfield network ซึ่งมีการเรียนรู้ทั้งแบบมีผู้ช่วยและไม่มีผู้ช่วย จากการค้นคว้าวิจัยในอดีตนั้น สามารถใช้ข่ายงานระบบประสาทในเรื่องของการสังเคราะห์ท่าทางการเดินของหุ่นยนต์ทั้งแบบใช้ข้อมูลในอดีตมาสอน(Off-line) และ แบบเรียนรู้ไปพร้อมกับการเดินของหุ่นยนต์(On-line) นอกจากนี้ยังสามารถใช้ในการควบคุมทั้งแบบการเดินแบบสถิตและการเดินแบบพลวัตในพื้นที่ผิวที่แตกต่างกันได้ ตัวอย่างงานวิจัยด้านนี้ได้แก่ งานของ Kitamura[26] ได้นำ Hopfield NN มาใช้ในการควบคุมการเดินของหุ่นยนต์ร่วมกับการควบคุมที่ใช้โมเดลของอินเวอร์เตอร์เพนดูลัม งานของ Salatian[27] ได้ใช้การเรียนรู้เชิงบังคับ(Reinforcement Learning) ในการสอนหุ่นยนต์ในการปรับท่าทางการเดินทั้งแบบ Offline และแบบ Online สำหรับการเดินบนพื้นเรียบและบนพื้นที่มีความชัน หุ่นยนต์ที่ใช้ทดสอบเป็นหุ่นยนต์ 8 แขน และมีอุปกรณ์วัดค่าแรงกระทำที่ฝ่าเท้าเพื่อใช้สอนหุ่นยนต์ โดยไม่จำเป็นต้องรู้แบบจำลองทางพลวัตของตัวหุ่นยนต์ Miller [8] ได้พัฒนาข่ายงานระบบประสาทแบบ CMAC เพื่อใช้ในการควบคุมแบบทันท่วงที(Real time control) ประกอบด้วยสามโครงข่าย กล่าวคือ โครงข่ายที่หนึ่งในการควบคุมการรักษาสมดุลด้านหน้าและหลัง โครงข่ายที่สองใช้ในการควบคุมสมดุลด้านซ้ายและหลัง และสามเพื่อใช้ในการเรียนรู้แบบจำลองของหุ่นยนต์แบบสถิต (Kinematics model) หลังจากทดสอบหุ่นยนต์สามารถเดินได้ดีในทางตรง แต่ยังไม่สามารถเดินแบบเปลี่ยนแปลงความเร็ว หรือเดินบนพื้นเอียงได้ จากนั้น Miller ได้พัฒนาต่อเป็นโครงข่ายเจ็ดส่วนและเพิ่มเติมอุปกรณ์วัดความเร่งและความเร็วเชิงมุมเข้าที่ตัวหุ่น ทำให้หุ่นยนต์สามารถเดินแบบเปลี่ยนแปลงความเร็วได้ ซึ่งแสดงให้เห็นว่าข่ายงานระบบประสาทแบบ CMAC มีความสามารถในการเรียนรู้ที่เร็วขึ้นและโครงสร้างซับซ้อนน้อยกว่าข่ายงานระบบประสาทแบบหลายชั้นเดิม แต่ก็มีข้อเสียที่ใช้หน่วยความจำจำนวนมาก

2.4.3 การใช้ตรรกศาสตร์คลุมเครือในการควบคุมหุ่นยนต์สองขา

มีนักวิจัยบางท่านที่นำตรรกศาสตร์คลุมเครือมาใช้ในการควบคุมหรือสังเคราะห์ท่าทางการเดินของหุ่นยนต์ ตัวอย่างเช่น Ching Long Shih [28,29,30] ได้นำเสนอการใช้ตรรกศาสตร์คลุมเครือในการควบคุมตำแหน่งของแรงที่กระทำที่ฝ่าเท้าในการเดินแบบสถิตศาสตร์ของหุ่นยนต์สองขา 12 องศาอิสระ Shuta Murakami [31] ในใช้ระบบควบคุมแบบฟัซซี่นี้ในการควบคุมการเดินของหุ่นยนต์ 7 แขนในสามารถเดินแบบพลวัตได้ M.Vukobratovic [1] ได้เสนอวิธีการนำตรรกศาสตร์คลุมเครือมาใช้ในการปรับค่าตัวแปรที่สำคัญของตัวควบคุมแบบ PID ที่ใช้ในการควบคุมการเดินของหุ่นยนต์สองขา และพิสูจน์ในเชิงความมีเสถียรภาพของระบบควบคุมแบบนี้ และ Shuta Murakami ยังได้กล่าวถึงปัญหาของการวิเคราะห์หาลักษณะการเคลื่อนที่ของการเดินแบบพลวัตของหุ่นยนต์สองขาให้คล้ายคนเป็นเพราะว่ารูปแบบของสมการทางคณิตศาสตร์มีความซับซ้อนมาก ในบทความนี้เราพิจารณาถึงรูปแบบการควบคุมของหุ่นยนต์สองขา (biped Robot) โดยใช้ตัวควบคุมแบบฟัซซี่มาใช้ในการควบคุมการเดินแบบพลวัตของหุ่นยนต์ (Dynamic Walking Robot) เริ่มจากเราจะสร้างตัวควบคุมแบบฟัซซี่ มาใช้ควบคุมแบบจำลองเดิมของหุ่นยนต์ จากนั้นเราจะหาตัวควบคุมแบบฟัซซี่ที่ดีที่สุดในการควบคุมหุ่นยนต์สองขานี้ และจำลองผลการทดลองบนคอมพิวเตอร์ ปรากฏว่าเป็นไปด้วยดี ทำให้เราเชื่อว่าการควบคุมการเดินแบบพลวัตของหุ่นยนต์ชนิดนี้โดยใช้ตัวควบคุมแบบฟัซซี่สามารถทำได้ แม้ว่าการเคลื่อนที่ในตอนต้นยังไม่ราบเรียบเท่าไรนัก ถ้าเราสามารถปรับเปลี่ยนหรือพัฒนาในส่วนของการสร้างรูปแบบการเดินอย่าง

เช่นใช้ Cost function เพื่อหาค่าต่ำที่สุดของการสร้างแนวทางการเคลื่อนที่ก็น่าจะทำให้การเคลื่อนที่ราบเรียบขึ้น เป็นต้น

2.4.4 การใช้กลไกเชิงพันธุกรรมในการการควบคุมหุ่นยนต์สองขา

ว่าด้วยเรื่องที่เกี่ยวข้องกับการเคลื่อนไหวหรือเคลื่อนที่ของหุ่นยนต์ กลไกเชิงพันธุกรรมใช้ในการหารูปแบบการเคลื่อนที่หรือเส้นทางการเคลื่อนที่ของแต่ละข้อต่อของหุ่นยนต์โดยพิจารณาถึงการใช้พลังงานที่เหมาะสมเป็นสำคัญ วิธีการในการหาเส้นทางการเคลื่อนที่นั้นแบ่งเป็นสองขั้นตอนใหญ่ คือ กลไกเชิงพันธุกรรมใช้ในการหาค่าพลังต่ำสุดของมอเตอร์ของหุ่น และกลไกเชิงวิวัฒนาการใช้ในการหารูปแบบท่าทางของหุ่นยนต์ ตัวอย่างเช่น M-Y Cheng [32] ได้กล่าวถึงการประยุกต์ใช้กลวิธีการทางพันธุกรรมในการควบคุมการเดินแบบพลวัตที่มีความซับซ้อนและยาก ซึ่งใช้วิธีทางพันธุกรรมนี้กับหลายๆเงื่อนไขของการเดินเช่น ทางเรียบ ทางชัน ก้าวการเดินที่ต่างกัน และความเร็วการเดินที่ต่างกัน พบว่ากลวิธีการพันธุกรรมนี้สามารถให้รูปแบบการควบคุมและลักษณะการเดินที่เหมาะสมกับทุกสภาวะที่ต้องการ T. Arakawa และคณะ[31] ได้ประยุกต์ใช้วิธีการทางพันธุกรรม (Genetic Algorithm) มาใช้ในการออกแบบเส้นทางการเดินของหุ่นยนต์ที่ดีที่สุดโดยพิจารณาถึงแง่ของความเร็วและความเร่งในแต่ละจุดของการเคลื่อนที่ ที่เหมาะสมทำให้หุ่นยนต์มีการเคลื่อนที่ที่ราบรื่นไม่เกิดการกระตุก มีเสถียรภาพ และยังรักษาความสมดุลของร่างกายซึ่งพิจารณาจาก ZMP ไว้ได้ นอกจากนี้ยังนำไปทดลองกับหุ่นยนต์ IWR III ซึ่งเป็นหุ่นยนต์ 8 แขน พบว่าสามารถเดินได้อย่างราบรื่นและมีเสถียรภาพ เป็นต้น

2.4.5 การใช้ระบบผสมในการในการการควบคุมหุ่นยนต์สองขา

ดังนั้นจะเห็นได้ว่าทั้งสองแบบของการคำนวณเชิงอ่อน(SC) อันได้แก่ข่ายงานระบบประสาทและตรรกศาสตร์ต่างก็สามารถประยุกต์ใช้กับการออกแบบท่าทางการเดินของหุ่นยนต์และควบคุมการเดินของหุ่นยนต์ให้เดินได้อย่างมีเสถียรภาพได้ แต่อย่างไรก็ตามก็ยังไม่สมบูรณ์ในตัวเอง จึงต้องมีการรวมทั้งสองแบบเข้าด้วยกันซึ่งเราเรียกว่า ระบบอัจฉริยะผสม (Hybrid Intelligent System) ซึ่งเป็นที่นิยมมากขึ้น เพราะว่าเราสามารถนำความรู้และเหตุผลเชิงบรรยายและข้อมูลเชิงตัวเลขที่ได้จากการสังเกตและทดลอง มาพัฒนาระบบและแก้ปัญหาที่สลับซับซ้อนได้ ตัวอย่างงานวิจัยได้แก่ Jih-Gau-Juang[33] ได้ทำการ ศึกษาวิจัยถึงการสร้างแบบแผนการเรียนรู้ลักษณะเส้นทางการเดินของหุ่นยนต์สองขาโดยใช้ Fuzzy modeling neural network แบบแผนการเรียนรู้สร้างลักษณะการเดินโดยกำหนดเส้นทางการเดินอ้างอิง อาทิเช่น ระยะการก้าวที่ต้องการ ความสูงในการยกขา และเวลาที่ใช้ในแต่ละก้าว แบบแผนที่นำเสนอนี้ใช้ตัวควบคุมแบบฟัซซี่ ร่วมกับแบบจำลองผกผันเชิงเส้นของหุ่นยนต์(linearized inverse biped model) และใช้ข่ายงานระบบประสาทหลายชั้นแบบป้อนไปข้างหน้า ในการควบคุมลักษณะการก้าวเดินของหุ่นยนต์โดยขั้นตอนการเรียนรู้ของข่ายงานระบบประสาทนี้ใช้กลวิธีป้อนกลับตามเวลา (Backpropagation through time) โดยแบบจำลองเชิงเส้นผกผันจะเป็นตัวส่งค่าความผิดพลาดไปให้ข่ายงานระบบประสาท ในงานวิจัยนี้นั้นได้จำลองผลการทดสอบกับหุ่นยนต์ห้าลึงค์บนคอมพิวเตอร์ พบด้วยแนวความคิดนี้สามารถควบคุมหุ่นยนต์ให้เดินไปตามเส้นทางที่กำหนดได้

2.5. แนวทางออกแบบระบบควบคุมที่ใช้ในการควบคุมหุ่นยนต์สอง

จากบทความที่ได้กล่าวมาข้างต้นเป็นการกล่าวถึงเฉพาะในแง่ของการนำข่ายงานระบบประสาทซึ่งมีความเหมาะสมในแง่ของความถูกต้อง แต่ใช้เวลาในการเรียนรู้มาก ซึ่งเป็นอุปสรรคต่อการควบคุมแบบทันที(Real-time control) ดังนั้นเพื่อแก้ปัญหาดังกล่าวจึงได้มีผู้คิดค้นวิธีการใหม่ขึ้นมา เรียกว่า “Soft-Computing” ซึ่งเป็นการนำเอาข้อดีของข่ายงานระบบประสาท (Neural Network) ระบบตรรกศาสตร์แบบคลุมเครือ (Fuzzy Logic System) และกลวิธีการถ่ายถอดพันธุกรรม (Genetic Algorithm) สิ่งที่ได้คือความสามารถในการควบคุมระบบซึ่งมีความไม่แน่นอนสูงได้ดีกว่าการควบคุมดั้งเดิม (Classical Control Approach) ตัวอย่างเช่น ระบบที่มีตัวแปรไม่ทราบค่าจำนวนมาก สรุปคือเราสามารถนำเอา Soft-Computing มาใช้ในการควบคุมหุ่นยนต์สองขา เพื่อให้สามารถเดินทรงตัวแบบพลวัติในสภาวะแวดล้อมต่างๆที่รู้ค่าและไม่รู้ค่าได้

สำหรับงานวิจัยนี้คือการหารูปแบบการเดินล่วงหน้าสำหรับหลบหลีกสิ่งกีดขวางที่อยู่เบื้องหน้าของหุ่นยนต์โดยดูจากการมองเห็น การออกแบบรูปแบบการเดินนี้ต้องรักษาเสถียรภาพการเดินแบบพลวัติภายใน step การเดินที่น้อยที่สุด เพื่อหลบหลีกสิ่งกีดขวางนั้น มันมีความเป็นไปได้ที่จะสร้างระบบที่มีเสถียรภาพทางพลวัติในการเคลื่อนที่ของหุ่นยนต์แบบทันเวลา(Real Time) ภายใต้สิ่งแวดล้อมที่ไม่แน่นอน และสิ่งกีดขวางที่ไม่รู้มาก่อน

มีข้อมูล 2 แบบ ที่ใช้ในการควบคุมหุ่นยนต์ แบบแรก คือ ข้อมูลเชิงตัวเลข(Numerical Data)ที่ได้จากการวัด เช่นจากอุปกรณ์วัดแรง (force sensor) อุปกรณ์วัดมุมของแต่ละข้อต่อ(Incremental - Encoder) และอุปกรณ์วัดมุมและความเร่งของตัวหุ่น(Inertial Sensor or Attitude & Heading Reference System) ส่วนแบบที่สองคือ ข้อมูลเชิงบรรยาย (Linguistic Information) จากการเดินของมนุษย์ และการศึกษาเรื่องชีวกลศาสตร์(Biomechanic) ซึ่งเราพบว่า ข้อมูลแบบใดแบบหนึ่งไม่สามารถครอบคลุมเรื่องการเดินของหุ่นยนต์ได้ทั้งหมด ดังนั้นเราจึงนำข้อมูลทั้งสอง แบบมาใช้ควบคู่กัน โดยข้อมูลเชิงบรรยายนั้นเราจะใช้เป็นข้อมูล สำหรับการควบคุมแบบฟัซซี่ โดยสร้างเป็นกฎของฟัซซี่ขึ้นมา (If – then rule) ส่วนข้อมูลเชิงตัวเลขนั้น เรานำไปใช้สำหรับข่ายงานระบบประสาท (NN) ซึ่งรวมเข้าเป็นระบบควบคุมแบบ Neuro-fuzzy การรวมกันแบบนี้เรียกระบบใหม่นี้ว่า Soft – computing ซึ่งน่าจะเหมาะสมกับการควบคุมหุ่นยนต์สองขา ซึ่งมีความไม่เป็นเชิงเส้นสูง (non-linear) ความไม่เสถียรภาพ (Unstable) มีการเปลี่ยนแปลงตามเวลา (time variant) และต้องปรับตัวตามสิ่งแวดล้อมรอบข้าง และรวมถึงการเดินภายใต้แรงโน้มถ่วงของโลกอีกด้วย

ในงานวิจัยของ Zhou .C [34] ได้เสนอวิธีการสังเคราะห์ท่าทางการเดินโดยการรวมข้อมูลเชิงบรรยายเข้ากับข้อมูลเชิงตัวเลข เพื่อใช้ในการปรับปรุงสมรรถนะการเดินแบบพลวัติ ยิ่งไปกว่านั้นวิธีการนี้ยังใช้ความรู้เบื้องต้น(Priori knowledge) ในการควบคุมหุ่นยนต์ ข้อเสียของวิธีการนี้ คือ รายละเอียดและความถูกต้องของข้อมูลเชิงตัวเลข เป็นเรื่องที่เป็นไปได้ยากสำหรับ หุ่นยนต์สองขา แนวทางใหม่ที่ Zhou C. ได้เสนอนั้น ได้นำการเรียนรู้แบบบังคับ (Reinforcement Learning) มาใช้สอนระบบแทน โดยใช้สัญญาณป้อนกลับเพียงค่าถูกหรือผิดหรือค่าเป็นช่วง ๆ เพื่อมาประเมินเสถียรภาพของหุ่นยนต์ ในงานวิจัยของเรานำการเรียนรู้แบบบังคับชนิดไม่ต้องมีผู้ช่วยสอน(Unsupervised Reinforcement Learning)

มาเป็นตัวปรับระบบสังเคราะห์ท่าทางการเดินแบบ Neuro- fuzzy ของหุ่นยนต์สองขา สำหรับในงานวิจัยด้านการนำ Reinforcement Learning มาใช้ปรับระบบควบคุมแบบฟัซซี่ หรือแบบข่ายงานระบบประสาทยังมีน้อย

ในการเดินของมนุษย์ท่าทางการเดิน(gait) ขึ้นอยู่กับการเคลื่อนไหวของส่วนขาและลำตัวประกอบกัน ส่วนลำตัวหุ่นยนต์จะเป็นส่วนที่หนักและมีความเฉื่อยสูง ถ้าส่วนลำตัวเคลื่อนไหวอย่างมีประสิทธิภาพ และมีเสถียรภาพ แนวโน้มที่การเดินของหุ่นยนต์สองขานี้ จะมีเสถียรภาพก็ย่อมมีสูง ในงานวิจัยที่ผ่านมาส่วนใหญ่เราใช้ลำตัวเป็นส่วนรักษาเสถียรภาพของหุ่นยนต์(Biped Stabilization)

จากที่กล่าวมาแล้วถึงเรื่องของความมีเสถียรภาพในการเดินของหุ่นจากการพิจารณาการเคลื่อนไหวของลำตัวเป็นหลัก และการสังเคราะห์ท่าทางการเดินล่วงหน้า(Prescribed Synergy method) เราจะนำเสนอวิธีการสังเคราะห์แนวใหม่ ที่ใช้การเรียนรู้แบบบังคับมาปรับตัวสังเคราะห์ท่าทางการเดินแบบ neuron-fuzzy โดยในขั้นเริ่มต้นเราจะออกแบบท่าทางการเดินส่วนขาล่วงหน้า และไม่มีการปรับระหว่างการเดิน โดยจะเน้นไปที่การเคลื่อนไหวของลำตัวเป็นหลัก [34,35,36,37,38] โดยเรานำเสนอวิธีการสังเคราะห์นี้สามารถใช้สำหรับการรักษาสมดุลแบบพลวัตของลำตัวทั้งในแนวด้านข้าง (Sagittal Plan) และแนวด้านหน้า (frontal Plan) ในบทความต่อไป

บทที่ 3 การออกแบบระบบนิวโรฟuzzy

จากที่ได้กล่าวมาแล้วในหัวข้อที่ผ่านมา สิ่งที่เราต้องการคือระบบนิวโรฟuzzyที่สามารถเรียนรู้ด้วยตัวเองและสามารถจดจำรูปแบบความ ไม่เป็นเชิงเส้นของระบบพลวัตทางกลต่างๆได้ จากการศึกษาค้นคว้าผลงานที่ผ่านมาทำให้สามารถสรุปได้ว่า ระบบนิวโรฟuzzyที่ควรออกแบบควรเริ่มจากระบบนิวโรฟuzzy Takagi-Sugeno-Kang ซึ่งเราสามารถสร้างระบบนี้จากข้อมูลทางตัวเลข (Numerical data) หรือจากข้อมูลเชิงบรรยาย (Knowledge base) ได้

ระบบนิวโรฟuzzyแบบ Takagi-Sugeno-Kang นั้น โดยพื้นฐานจะมีจำนวนกฎน้อย ก็สามารถบรรยายความซับซ้อนของระบบพลวัตที่ไม่เป็นเชิงเส้นได้ และน้อยกว่าระบบฟuzzyแบบ Mamdani [39,40] และสำหรับ ระบบนิวโรฟuzzyแบบลำดับที่ 1 มีรูปแบบสมการสำหรับ K กฎ ดังต่อไปนี้

$$R_i : \text{If } x_1 \text{ is } A_{i1} \text{ and } x_2 \text{ is } A_{i2} \text{ and } \dots \text{ and } A_{in} \\ \text{then } y_i = \bar{a}_i \bar{x} + b_i, \quad i = 1, 2, \dots, K \quad (3.1)$$

เมื่อ R_i คือ กฎข้อที่ i

$x_1, x_2, \dots, x_n$ คือ ชุดข้อมูลขาเข้า (Input Variable)

$A_{i1}, A_{i2}, \dots, A_{in}$ คือ ฟuzzyเซตที่เกี่ยวข้องกับข้อมูลขาเข้า

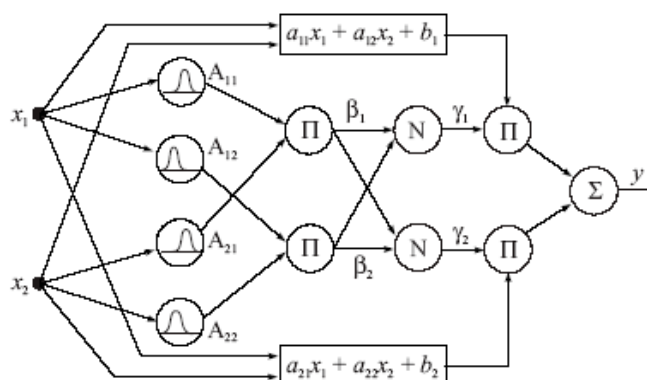
y_i คือ ข้อมูลขาออก ซึ่งได้จากสมการเชิงเส้นในแต่ละกฎ

a_i, b_i คือ ค่าคงที่ของส่วน consequent ในกฎของฟuzzy

ตัวอย่างเช่นระบบนิวโรฟuzzyที่มี ตัวแปร และ กฎ สมการที่ 3.1 จะเป็นดังนี้

$$R1 : \text{If } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{21} \\ \text{then } y_1 = a_{11}x_1 + a_{12}x_2 + b_1 \\ R2 : \text{If } x_1 \text{ is } A_{21} \text{ and } x_2 \text{ is } A_{22} \\ \text{then } y_2 = a_{21}x_1 + a_{22}x_2 + b_2 \quad (3.2)$$

และมีโครงสร้างข่ายงานดังรูปที่ 3.1



รูปที่ 3.1 แสดงตัวอย่าง ระบบนิวโรฟuzzy แบบ Takagi-Sugeno ลำดับที่ 1 สำหรับ 2 กฎ

สำหรับสมการระหว่างข้อมูลขาเข้าและขาออก คือ

$$y = \frac{\sum_{i=1}^K \beta_i(\vec{x}) y_i}{\sum_{i=1}^K \beta_i(\vec{x})} = \frac{\sum_{i=1}^K \beta_i(\vec{x}) (a_i^T \vec{x} + b_i)}{\sum_{i=1}^K \beta_i(\vec{x})} \quad (3.3)$$

$$= \sum_{i=1}^K \gamma_i(\vec{x}) (a_i^T \vec{x} + b_i)$$

เมื่อ $\beta_i(x)$ คือ degree of fulfillment ของกฎข้อที่ i และ $\gamma_i(\vec{x})$ เป็นการ normalize ของ $\beta_i(x)$:ซึ่งเราเรียกค่านี้ว่า basis function expansion

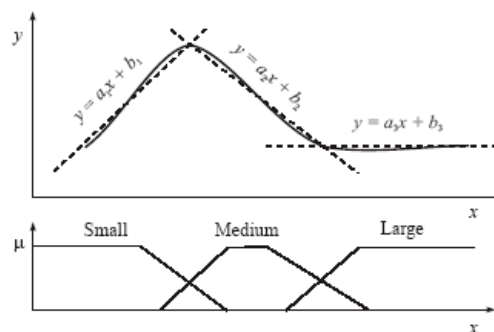
ในการเลือกสมการสมาชิก(Membership function) เราควรเลือก สมการที่มีความต่อเนื่อง (Continuous function) เพื่อสะดวกในการใช้กรรมวิธีทางคณิตศาสตร์เพื่อหาค่าที่เหมาะสมที่สุด ในที่นี้ เราเลือกสมการ Gaussian ซึ่งมีลักษณะสมการ ดังต่อไปนี้

$$u_{A_{ij}}(x_j, c_{ij}, \sigma_{ij}) = \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right) \quad (3.4)$$

เมื่อ ค่า c_{ij}, σ_{ij} ซึ่งเป็นค่าคงที่สำหรับสมการ Gaussian นั้น เราต้องทำการหาค่าและเมื่อเขียน ในรูปของค่า degree of fulfillment จะได้ค่า γ_i^j ดังนี้

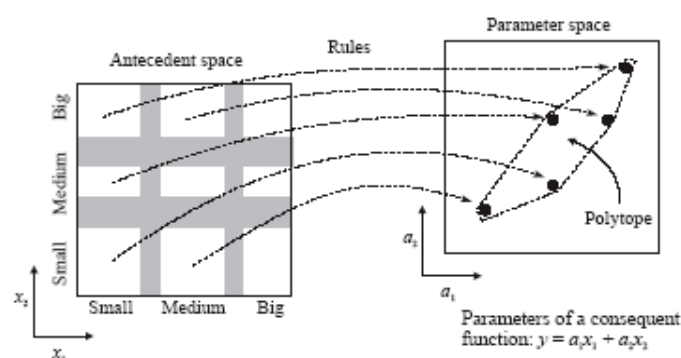
$$\gamma_i(\vec{x}) = \frac{\prod_{j=1}^p \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right)}{\sum_{i=1}^K \prod_{j=1}^p \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right)} \quad (3.5)$$

จากสมการข้างต้นที่ได้กล่าวมาแล้วนั้น ทำให้เราสามารถออกแบบส่วน antecedent หรือส่วน หน้าของกฎเงื่อนไขให้มีส่วนทับกัน (Overlap) ทำให้ระบบนิวโรฟัซซี่แบบนี้เหมาะกับการประมาณค่า ฟังก์ชัน ของระบบพลวัตได้ดี ดังรูปที่ 3.2 เห็นได้ว่าสามารถใช้ประมาณค่าสมการไม่เป็นเชิงเส้นได้เป็นอย่างดี



รูปที่ 3.2 แสดงการประมาณสมการไม่เป็นเชิงเส้นของระบบนิวโรฟัซซี่

ในกรณีของ T-S นี้ จะเปลี่ยนจากการพื้นที่ของข้อมูลขาเข้า (antecedent (input) space) ไปเป็นพื้นที่ของตัวแปรแบบเชิงเส้นแทน (quasi-linear system) แทน ดังรูปที่ 3.3



รูปที่ 3.3 แสดงตัวอย่างระบบนิเวศที่แบบ Takagi-Sugeno ที่เปลี่ยนจาก input space ไปเป็น parameter space

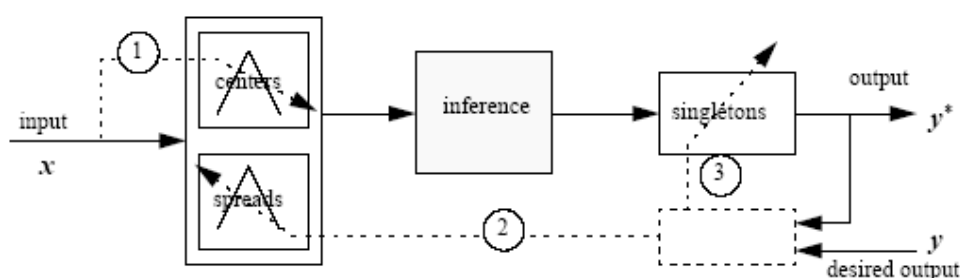
3.1 กลวิธีการสร้างระบบนิเวศที่แบบ Takagi-Sugeno

ในการออกแบบระบบนิเวศที่แบบ Takagi-Sugeno ให้สามารถ สร้างตัวเองได้เองและสามารถปรับเปลี่ยนตัวไปตามข้อมูลขาเข้าและขาออก รวมถึงสามารถนำไปใช้แบบ online-learning เราสรุปขั้นตอนสำคัญได้ 3 ขั้นตอนด้วยกันคือ

1. การสร้างกฎขึ้นมาอย่างอัตโนมัติในตอนเริ่มต้น (Initial Structure Identification)
2. การหาค่าพารามิเตอร์ของระบบในส่วน consequent ของแต่ละกฎ
3. การปรับโครงสร้างของระบบนิเวศที่ ว่าควรจะเพิ่มหรือลดจำนวนกฎ

ในขบวนการที่ 1 และ 2 นั้นเป็นขบวนการที่เรียกว่า ขบวนการเรียนรู้ (Learning Process) ซึ่งกระทำทั้งสองรูปแบบก็คือ ใช้การเรียนรู้แบบมีผู้ช่วยสอน โดยใช้ข้อมูลทางคณิตศาสตร์ ซึ่งเป็นการเรียนรู้โดยใช้ข้อมูลในอดีต (Off-line training) และ การเรียนรู้ที่อาศัยข้อมูลในปัจจุบัน ซึ่งเรียกว่าการเรียนรู้แบบทันที (Online training) รูปแบบกระบวนการเรียนรู้ในขั้นที่ 1 และ 2 ดังรูปที่ 3.4

ส่วนในขบวนการที่ 3 นั้น เป็นขบวนการที่อาศัยการเรียนรู้โดยใช้ข้อมูลในปัจจุบัน มาเป็นตัวพิจารณาว่าควรมีการปรับปรุงโครงสร้างของระบบนิเวศที่อย่างไร ในที่นี้คือขบวนการเพิ่มหรือลดจำนวนกฎนั่นเอง

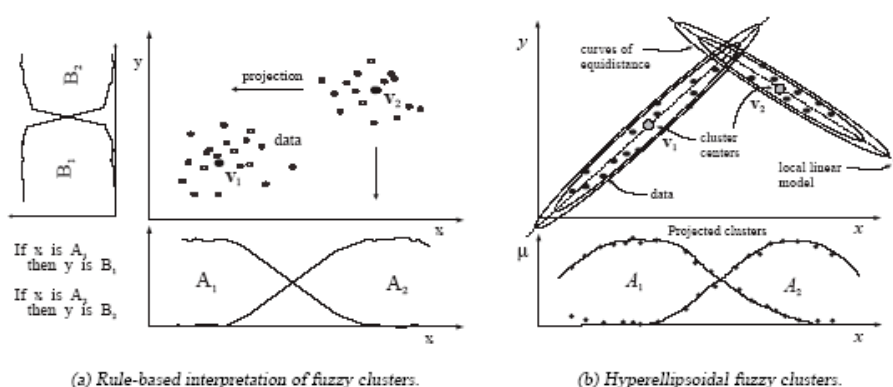


รูปที่ 3.4 แสดงตัวอย่างกระบวนการเรียนรู้ของระบบนิเวศที่แบบ Takagi-Sugeno

3.1.1 การออกแบบโครงสร้างและกฎของระบบนิวโรฟัซซีเมื่อเริ่มต้น

นอกจากการอาศัยความรู้ของผู้เชี่ยวชาญในการออกแบบจำนวนข้อมูลขาเข้า ข้อมูลขาออกและจำนวนกฎที่เกี่ยวข้องกันในแต่ละข้อมูล ซึ่งทำให้ระบบฟัซซีที่นำไปใช้งานได้แต่ก็ไม่เสมอไปที่บางครั้งกฎที่ออกแบบนั้นก็ไม่ถูกต้อง ดังนั้นต้องมีการปรับเปลี่ยนซึ่งทำให้ต้องเสียเวลานานในการออกแบบระบบฟัซซีให้สอดคล้องกับระบบพลวัต ในงานวิจัยนี้จึงมุ่งเน้นไปที่การประยุกต์ความรู้ของผู้เชี่ยวชาญและข้อมูลที่สามารถหาได้ของระบบนั้นๆ มาทำการสร้างเป็นระบบนิวโรฟัซซีแบบอัตโนมัติ โดยการเลือกจำนวนข้อมูลขาเข้าและออกนั้นจะเป็นหน้าที่ของผู้เชี่ยวชาญ ส่วนการหาจำนวนฟังก์ชันสมาชิก (Membership function) ต่อข้อมูลขาเข้า และจำนวนกฎที่สอดคล้องกับข้อมูลขาเข้าแต่ละตัวนั้นจะถูกสร้างขึ้นเองโดยพิจารณาจากข้อมูลขาเข้าและออกของระบบที่เราสนใจ ในที่นี้เราใช้ฟังก์ชันสมาชิกแบบ Gaussian เป็นแบบเริ่มต้น ขบวนการในการเลือกว่าแต่ละข้อมูลขาเข้าจะมีจำนวนฟังก์ชันสมาชิกกี่ตัวนั้น เราอาศัยหลักการแบ่งกลุ่มช่วงข้อมูลโดยใช้วิธีการ clustering

วิธีการแบ่งกลุ่ม(clustering) นิยมใช้มานานในเรื่องของการจำแนกวิเคราะห์ข้อมูล(data analysis) และการรู้จำรูปแบบ(pattern recognition) ซึ่งเป็นที่นิยมกันในข่ายงานระบบประสาท(Neural network) จุดประสงค์ของการแบ่งกลุ่มก็เพื่อแยกกลุ่มข้อมูลจำนวนมากออกเป็นกลุ่มย่อยโดยอาศัยหลักความคล้ายคลึงกัน(similarity)ซึ่งสามารถคำนวณได้จากวิธีการวัดระยะของข้อมูลที่เหมาะสม ทำให้สามารถแบ่งกลุ่มข้อมูลโดยพิจารณา ระยะทางที่ใกล้เคียงกันของแต่ละข้อมูลแนวคิดของการแบ่งกลุ่มข้อมูล ของระบบฟัซซี(Fuzzy Clustering) แสดงดังรูปที่ 3.5 (a) จากรูปแสดงให้เห็นถึงข้อมูลถูกแบ่งออกเป็นสองกลุ่มย่อย คือ กลุ่ม v_1 และ กลุ่ม v_2 โดยใช้วิธีการวัดระยะ (Euclidean distance measure) แต่ละกลุ่มข้อมูล โดยแบ่งออกเป็นส่วนของระบบฟัซซี (fuzzy partition matrix) ออกเป็นกลุ่มๆ v_i ซึ่งแต่ละข้อมูล $[x_i, y_i]$ ถูกอธิบายในรูปของดีกรีของฟังก์ชันสมาชิก



รูปที่ 3.5 กระบวนการออกแบบโครงสร้างของฟัซซีโดยใช้วิธีการแบ่งกลุ่ม

กฎเงื่อนไขของฟัซซี สามารถที่จะสร้างขึ้นโดยพิจารณาเป็นส่วนๆตามกลุ่มที่แยกกัน ในรูปที่ 3.5 (a) แสดงถึงกลุ่มข้อมูลที่แบ่งแยกเป็นสองกลุ่มและกฎที่สร้างออกมาเป็นสองเช่นกัน แนวคิดของความคล้ายคลึงนั้นแบ่งกลุ่มข้อมูลโดยอาศัยวิธีการวัดระยะ ตัวอย่างเช่นวิธีการวัดแบบเชิงเส้น (linear space,

Bezdek 1981) ที่เรียกว่า fuzzy c-means หรือการวัดแบบวงรี(ellipsoids, Gustafson and Kessel 1979) ซึ่งวิธีการนี้ดังรูปที่ 3.5 (b) วิธีการทั้งสองมีข้อจำกัดตรงที่ว่าต้องกำหนดจำนวนกลุ่มข้อมูลทำให้ยังเกิดความผิดพลาดได้ง่ายในกรณีที่ผู้เชี่ยวชาญมีประสบการณ์ไม่เพียงพอ ส่วนวิธีการที่นำมาใช้ในงานวิจัยนี้คือวิธี **subtracting** [chi94] ซึ่งเป็นวิธีการใช้จำแนกกลุ่มข้อมูลที่รวดเร็วและแบ่งกลุ่มข้อมูลได้เองโดยไม่ต้องกำหนดจำนวนกลุ่มไว้ล่วงหน้า และที่ทำได้รวดเร็วเนื่องจากอาศัยการอ่านข้อมูลทั้งหมดเพียงครั้งเดียว จึงเป็นวิธีที่เหมาะสมสำหรับใช้ในการหาจำนวนสมาชิกฟังก์ชัน(Membership function) และจำนวนกฎเงื่อนไขที่สอดคล้องของระบบนิวโรฟัซซี่ในตอนเริ่มต้นและสามารถนำไปใช้วิธีการหากลุ่มข้อมูลที่เหมาะสมที่สุดต่อไปได้เช่น FCM เป็นต้น

Subtractive Clustering เป็นวิธีการการแบ่งกลุ่มข้อมูลที่ใช้จุดข้อมูลมาสร้างจุดกึ่งกลางของกลุ่มแทนวิธีการเดิมที่ใช้ตำแหน่งในพื้นที่ของกลุ่มข้อมูล โดยเริ่มจากการพิจารณากลุ่มข้อมูลซึ่งสมมุติว่ามี n ตัว $\{x_1, x_2, x_3, \dots, x_n\}$ ในพื้นที่ M มิติ และถือว่าแต่ละจุดข้อมูลสามารถเป็นศูนย์กลางของกลุ่มข้อมูลย่อยๆได้ วิธีการในการเลือกจุดข้อมูลเพื่อมาหาตำแหน่งศูนย์กลางใช้วิธีวัดความหนาแน่นของแต่ละข้อมูล x_i

$$D_i = \sum_{j=1}^n \exp\left(-\frac{\|x_i - x_j\|^2}{(r_a/2)^2}\right), \quad (3.6)$$

เมื่อ r_a แทนค่าคงที่บวก ที่ใช้เป็นรัศมีในการวัดกลุ่มย่อยแต่ละกลุ่ม

หลังจากที่ทำการวัดความหนาแน่นของกลุ่มข้อมูล ค่าของข้อมูลใด ที่ทำให้ค่าความหนาแน่นสูงสุด ก็จะถูกเลือกเป็นศูนย์กลางของกลุ่มข้อมูลที่หนึ่ง ให้แทนข้อมูลตัวนั้นว่า x_{c1} และ D_{c1} แทนค่าความหนาแน่นของมัน ขั้นตอนต่อมาคือการวัดค่าความหนาแน่นใหม่โดยใช้สมการนี้แทน

$$D_i = D_i - D_{c1} * \exp\left(\frac{-\|x_i - x_{c1}\|^2}{(r_b/2)^2}\right), \quad (3.7)$$

เมื่อ r_b เป็นค่าคงที่บวก

ดังนั้นจุดข้อมูลที่อยู่ใกล้กับศูนย์กลางข้อมูลในกลุ่มข้อมูลชุดที่หนึ่งก็จะถูกลดความสำคัญลงหรือมีค่าความหนาแน่นน้อยลงนั่นเอง เมื่อพิจารณาค่าความหนาแน่นของข้อมูลใหม่ก็จะได้ศูนย์กลางข้อมูลชุดถัดไป ค่าคงที่ r_b โดยส่วยใหญ่มีค่า $1.5r_a$

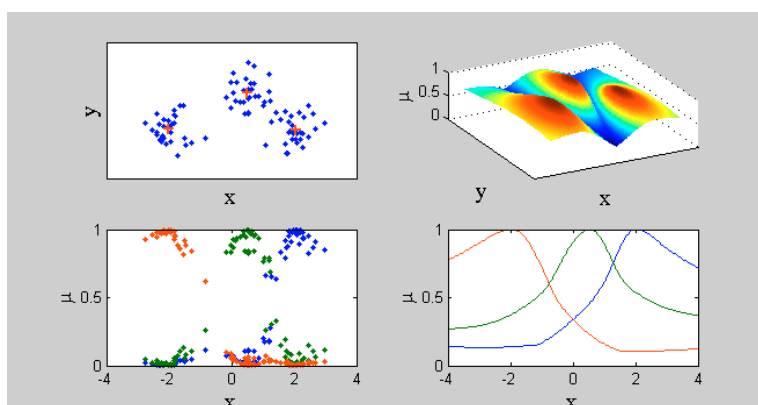
ขบวนการนี้กระทำซ้ำจนได้ค่าศูนย์กลางที่ครอบคลุมข้อมูลทั้งหมด หลังจากนั้นนำค่าศูนย์กลางของแต่ละข้อมูลไปใช้เป็นศูนย์กลางของแต่ละกฎในของระบบฟัซซี่ เราให้แต่ละศูนย์กลางคือ c_i ใน M มิติ โดยแต่ละ c_i นั้นจะมีสององค์ประกอบคือ p_i เป็นส่วนของข้อมูลขาเข้าและ q_i เป็นส่วนของข้อมูลขาออก ดังนั้นเราสามารถเขียนแต่ละข้อมูลอินพุตในรูปของจำนวนฟัซซี่ (degree of fulfillment) ดังนี้

$$\mu_i = \exp\left(\frac{-\|x - p_i\|^2}{(r_a/2)^2}\right) \quad (3.8)$$

ตัวอย่างเช่นถ้าแบ่งกลุ่มข้อมูลเป็นสามกลุ่ม เราสามารถเขียนส่วนฟังก์ชันสมาชิกของส่วนข้อมูลขาเข้า(Antecedent part) และตัวแปรต่างๆในส่วนหลังของกฎ (Consequent part) ของระบบนิวโรฟัซซี่ Takagi-Sugeno-Kang ได้ดังนี้

$$\begin{aligned} \text{If } x \text{ is } A_1 \text{ then } y &= a_1x + b_1, \\ \text{If } x \text{ is } A_2 \text{ then } y &= a_2x + b_2, \\ \text{f } x \text{ is } A_3 \text{ then } y &= a_3x + b_3 \end{aligned} \quad (3.9)$$

โดยแต่ละกลุ่มจะแสดงหนึ่งกฎในระบบนิวโรฟัซซี่ Takagi-Sugeno ค่าฟังก์ชันสมาชิกสำหรับเซต A_1 A_2 และเซต A_3 จะถูกสร้างโดยใช้สมการที่ 3.8 ข้างต้น และเป็นดังรูปที่ 3.6 ซึ่งแสดงให้เห็นถึงการแบ่งกลุ่มข้อมูลเป็นสามกลุ่ม จำนวนค่าสมาชิกฟังก์ชันแต่ละกลุ่ม ส่วนค่าตัวแปร a_i, b_i (Consequent parameter) สามารถปรับค่าให้ถูกต้องโดยใช้วิธีการประมาณค่าเหมาะสมเช่นวิธีประมาณค่าแบบ least square



รูปที่ 3.6 รูปแบบโครงสร้างของฟัซซี่โดยใช้วิธีการแบ่งกลุ่มโดยออกเป็นสามกลุ่ม

3.1.2 การปรับค่าพารามิเตอร์ส่วนเอาต์พุต (Least – square estimation of consequents)

เป็นวิธีในการประมาณค่าตัวแปรที่สนใจวิธีการหนึ่ง ซึ่งให้ผลดีกว่าการใช้วิธีการ Gradient descent โดยตัวแปรในส่วน consequent ที่ต้อง การปรับค่าคือ a และ b โดยใช้ข้อมูลขาเข้าและออกของระบบ $\{(x_i, y_i) | i = 1, 2, 3, \dots, N\}$ โดยให้ $x_i \in \mathcal{R}^P$ เป็นเวกเตอร์ของข้อมูลขาเข้า และ y_i เป็นข้อมูลขาออก ดังนั้น $X \in \mathcal{R}^{N \times p}$ คือเมตริกซ์ที่มีค่า x_k^T อยู่ในแถวและ $y \in \mathcal{R}^N$ เป็นเวกเตอร์ของค่าเอาต์พุต y_k มีรูปแบบสมการดังนี้

$$X = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N]^T \quad (3.10)$$

$$y = [y_1, y_2, \dots, y_N]^T \quad (3.11)$$

ดังนั้นในส่วนการตีความเอาต์พุตของระบบฟัซซี่ (Defuzzification) ดังแสดงในสมการที่ 3.3 เราสมมุติให้ค่า $\Gamma_i \in \mathcal{R}^{N \times N}$ เป็นเมตริกซ์เฉียง ของค่าค่าฟังก์ชันสมาชิกแบบปกติ $\gamma_i(x_k)$ ในสมการที่ 3.4

ที่มี k ค่า และสำหรับค่า X เราเพิ่มค่าอีกหนึ่งแถวเมตริกซ์ จะได้ $X_e = [X \ 1]$ นอกจากนี้ยังให้ X' เป็นเมตริกซ์ขนาด $\mathcal{R}^{N \times KN}$ โดยเป็นผลมาจากการคูณเมตริกซ์ (product) ระหว่างเมตริกซ์ Γ_i และเมตริกซ์ X_e เราจะได้สมการของ X' ดังนี้

$$X' = [\Gamma_1 X_e, \Gamma_2 X_e, \dots, \Gamma_K X_e] \quad (3.12)$$

สามารถเขียนตัวแปรในส่วนของ Consequent ในรูปของเวกเตอร์ของ $\theta \in \mathcal{R}^{K(p+1)}$ ดังนี้

$$\theta = [a_1^T, b_1, a_2^T, b_2, \dots, a_k^T, b_k]^T \quad (3.13)$$

ให้ข้อมูล X, Y จากสมการที่ 3.3 เขียนในรูปเมตริกซ์ $y = X'\theta + \varepsilon$ จากสมการนี้เราสามารถหาค่าตอบของค่า θ ได้ว่า

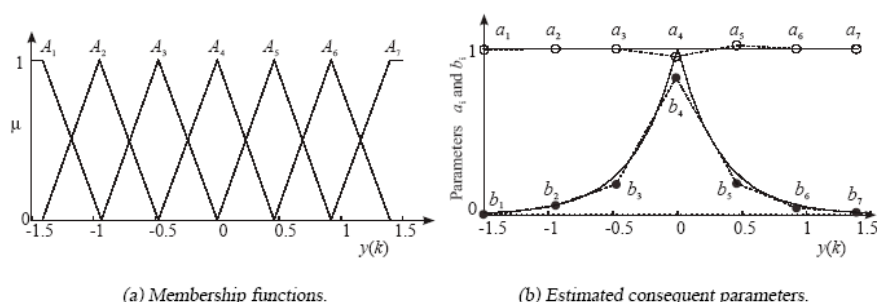
$$\theta = [(X')^T X']^{-1} (X')^T y \quad (3.14)$$

สมการที่ 3.14 นี้เป็นสมการการหาค่าที่เหมาะสมแบบน้อยสุด (Optimal least square) ซึ่งเหมาะกับโมเดลระบบแบบที่ใช้ทำนายค่า แต่วิธีการนี้จะทำให้ระบบนิวโรฟัซซี่ที่ได้เหมาะกับการหาค่าแบบเฉพาะที่ (local model estimator) ถ้าเราต้องการให้ถูกต้องมากยิ่งขึ้นเราก็ใช้รูปแบบการประมาณค่าด้วยวิธีถ่วงน้ำหนัก (Weight least squares approach) สำหรับแต่ละกฎของฟัซซี่

ดังสมการ

$$[a_i^T, b_i]^T = [X_e^T \Gamma_i X_e]^{-1} X_e^T \Gamma_i y \quad (3.15)$$

ในกรณีนี้ ค่าตัวแปรในส่วน Consequent ของแต่ละกฎจะแยกจากกันอิสระ ดังตัวอย่างในรูปที่ 3.7 เป็นระบบฟัซซี่ที่มีฟังก์ชันสมาชิกเป็นรูปสามเหลี่ยม จำนวน 7 รูป นั่นคือจะมีกฎเงื่อนไข 7 กฎ และมีค่าตัวแปรในส่วน consequent ทั้งหมด 14 ตัว ($a_1, a_2, \dots, a_7, b_1, b_2, \dots, b_7$) โดยค่าเส้นปะในรูปที่ 3.7 (b) เป็นการประมาณค่าโดยใช้ least square



รูปที่ 3.7 (a) แสดงตัวอย่างฟังก์ชันสมาชิก

(b) การประมาณค่าตัวแปร a และ b ของระบบฟัซซี่นี้

3.1.3 การปรับโครงสร้างของระบบนิวโรฟัซซี่ (Self-generation procedure)

ขบวนการนี้เป็นขบวนการในการเพิ่มหรือลดจำนวนกฎเงื่อนไข รวมถึงตัวแปรในส่วน Consequence ของระบบนิวโรฟัซซี่เดิมที่ถูกสร้างขึ้นในขั้นตอนที่หนึ่งและสอง โดยความสำคัญของชั้น

ตอนนี้อยู่ที่ความเร็วในการเรียนรู้ เพื่อที่นำระบบดังกล่าวไปใช้ในการเรียนรู้แบบทันที (Online - learning) โดยมีขั้นตอนสำคัญสามขั้นตอนคือ

- 1) ค้นหาข้อมูลชุดใหม่ที่เข้าสู่ระบบว่าอยู่ในกลุ่มข้อมูลชุดไหน
- 2) คำนวณหาค่าสมาชิกฟังก์ชันที่เหมาะสมกับข้อมูลชุดนั้น และ
- 3) ทำการปรับค่าตัวแปรในส่วน consequent ของระบบนิวโรฟuzzy เพื่อให้สอดคล้องกับข้อมูลชุดใหม่

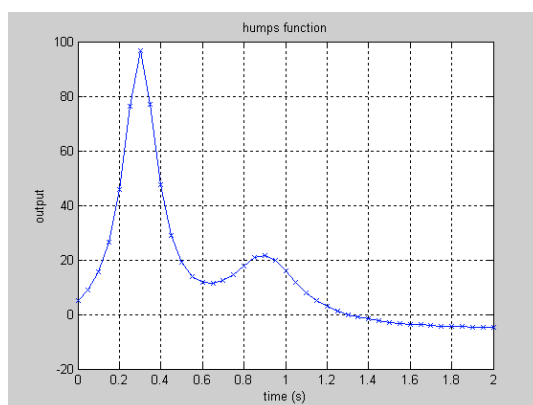
3.2. ตัวอย่างการนำนิวโรฟuzzyที่ออกแบบมาใช้งาน

3.2.1 การประมาณค่าฟังก์ชันแบบไม่เป็นเชิงเส้น

การประมาณค่าฟังก์ชันด้วยการใช้หนึ่งอินพุตและหนึ่งเอาต์พุตโดยใช้นิวโรฟuzzy TSK โดยระบบที่ยกตัวอย่างเป็นแบบสถิติศาสตร์ซึ่งเป็นส่วนประกอบของชาเยนน์เพื่อแสดงความเป็นสมการไม่เป็นเชิงเส้น

$$y = \text{humps}(x), \quad x = 0, 0.01, 2 \quad (3.16)$$

สมการประกอบด้วยข้อมูลตัวอย่างของคู่ข้อมูลขาเข้าและออก (x, y) ทั้งหมด 101 ข้อมูล มีลักษณะดังรูป 3.8

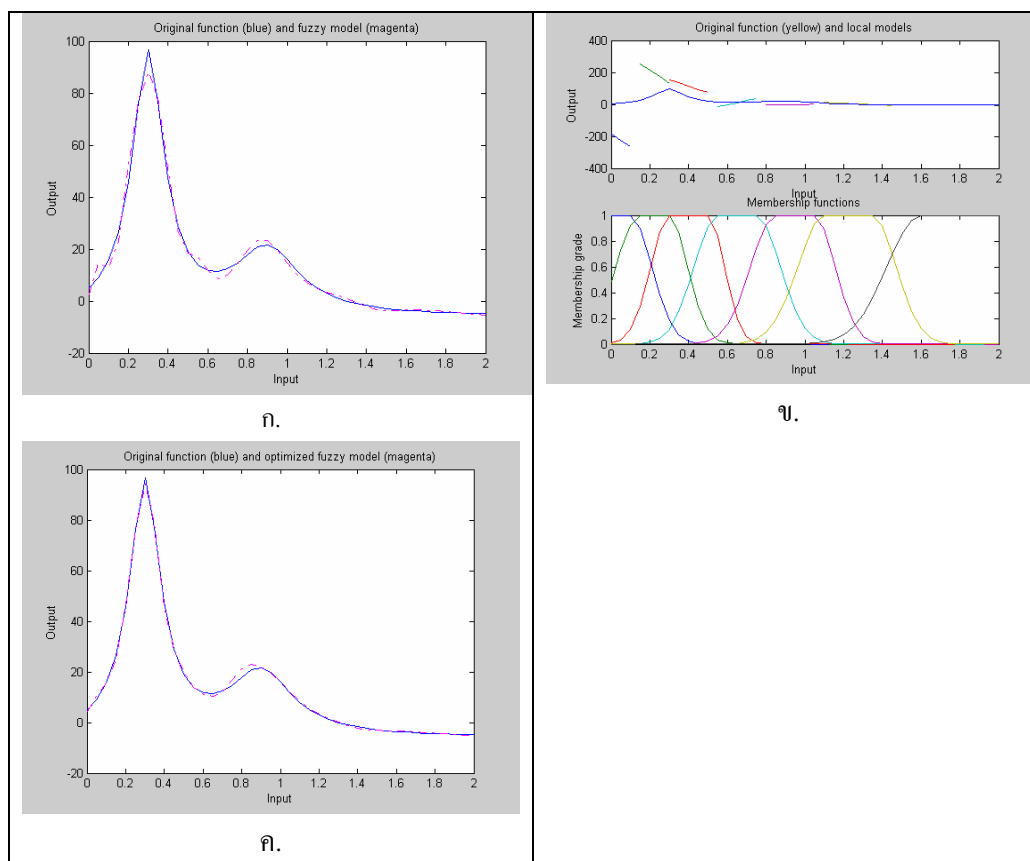


รูปที่ 3.8 แสดงตัวอย่างกราฟระหว่างข้อมูลขาเข้าและข้อมูลขาออก y

จากข้อมูลตัวอย่างที่มี เราใช้กระบวนการที่ หนึ่งและสองในหัวข้อที่ผ่านมาการใช้วิธี subtractive clustering และการปรับค่าตัวแปรของส่วน consequent ดังตัวอย่างกฎของระบบนิวโรฟuzzy TSK ในสมการที่ 3.9 ดังนี้

$$\begin{aligned} \text{If } x \text{ is } A_1 \text{ then } y &= a_1x + b_1, \\ \text{If } x \text{ is } A_2 \text{ then } y &= a_2x + b_2, \end{aligned}$$

หลังจากผ่านการเรียนรู้ระบบนิวโรฟuzzyที่ได้ มีลักษณะดังรูปที่ 3.9 แสดงถึงความสามารถในการประมาณค่าฟังก์ชันแบบไม่เป็นเชิงเส้น และจำนวนกฎและสมาชิกที่ถูกสร้างขึ้น



รูปที่ 3.9 ก. แสดงกราฟข้อมูลขาเข้าและข้อมูลขาออก y หลังใช้ subtractive clustering

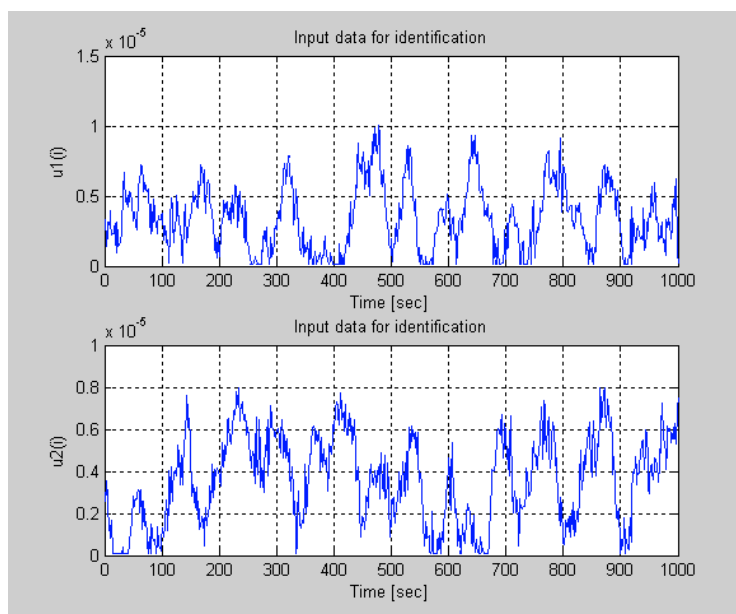
ข. แสดงค่าฟังก์ชันสมาชิกและช่วงของการแบ่งกลุ่มข้อมูล

ค. แสดงกราฟระหว่างข้อมูลขาเข้าและข้อมูลขาออก y หลังปรับค่า Consequent parameter

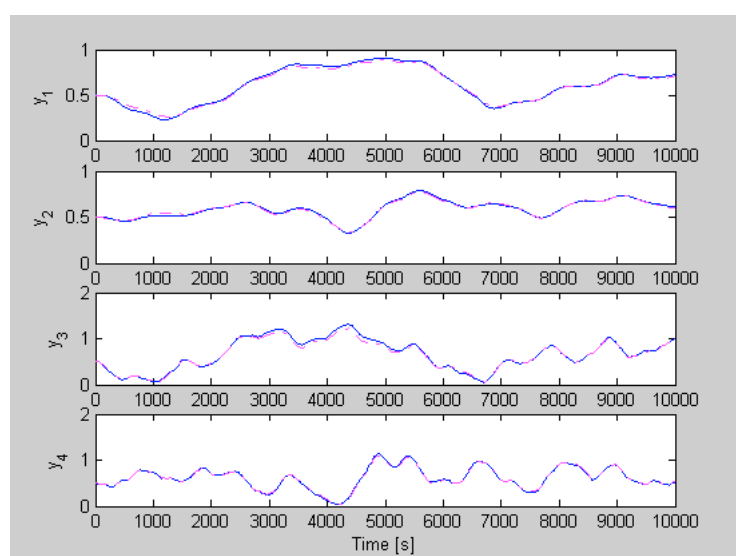
จากรูปที่ 3.9 แสดงให้เห็นว่าระบบนิวโรฟัซซีมีความสามารถในการประมาณค่าสมการไม่เป็นเชิงเส้นได้เป็นอย่างดีดังในรูป ก. ส่วนในรูป ข. แสดงให้เห็นว่าข้อมูลถูกแบ่งออกเป็นช่วงๆ ทั้งหมด 7 ช่วง และระบบนิวโรฟัซซีนี้ก็จะสร้างกฎทั้งหมด 7 กฎเพื่อรองรับ หลังจากนั้นก็จะปรับค่าพารามิเตอร์ ในส่วนหลังของกฎ (Consequent) โดยวิธี least square ก็จะได้ผลตามรูปที่ ค.

3.2.2 การประมาณค่าระบบหลายอินพุตและหลายเอาต์พุต (Modeling a MIMO Dynamic System)

ข้อมูลที่นำมาทดสอบเป็นข้อมูลที่มีจำนวนอินพุต 2 ค่า และมีจำนวนเอาต์พุต 4 ค่า จำนวนชุดข้อมูลทั้งหมด 1000 ชุด ดังรูปที่ 3.10 แสดงชุดข้อมูลอินพุตของระบบที่นำมาใช้



รูปที่ 3.10 แสดงตัวอย่างกราฟชุดข้อมูลขาเข้า u_1 และ u_2



รูปที่ 3.11 แสดงการเปรียบเทียบระหว่างชุดข้อมูลขาออกและที่ได้จากระบบนิวโรฟัซซี่

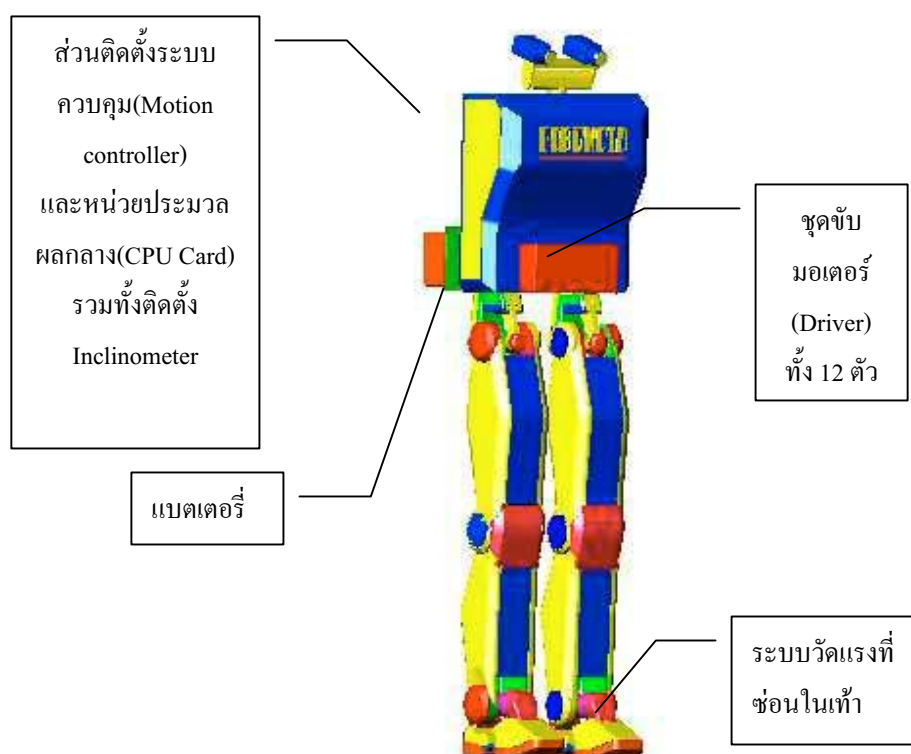
จากรูปที่ 3.11 แสดงให้เห็นว่าระบบนิวโรฟัซซี่ที่ออกแบบมีความสามารถในการเรียนรู้ระบบหลายข้อมูลอินพุตและและข้อมูลเอาต์พุตได้ดี และที่สำคัญคือเวลาในการเรียนรู้ที่รวดเร็วทำให้เราสามารถนำระบบดังกล่าวนี้ไปใช้ในการเรียนรู้แบบทันที (Online training) ได้

จากการออกแบบระบบที่กล่าวมาข้างต้น และผลการทดลองนำระบบนิวโรฟัซซี่นี้ไปทดสอบกับระบบอินพุต-เอาต์พุตทั้งแบบข้อมูลเดียวและหลายข้อมูล พบว่าระบบนิวโรฟัซซี่นี้มีความสามารถในการจำลองระบบหรือหาความสัมพันธ์ระหว่างข้อมูลได้เป็นอย่างดี ในบทต่อไปจะได้กล่าวถึงการนำระบบนิวโรฟัซซี่นี้ไปใช้ต่อไป

บทที่ 4 หุ่นยนต์สองขาที่ใช้ในการทดลอง

4.1 ด้านระบบกลของหุ่นยนต์สองขาของ FIBO

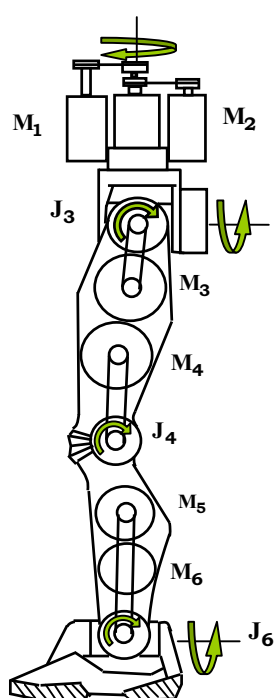
ระบบทางกลของหุ่นยนต์สองขา ที่ใช้ในการพัฒนาระบบการใช้ระบบการคำนวณเชิงอ่อน (Soft – Computing) ในการสร้างรูปแบบการเดินแบบพลวัตของหุ่นยนต์ฮิวแมนนอยด์ เป็นผลงานออกแบบของนักศึกษาปริญญาโทโครง การวิจัย Image Guided Motion of a Humanoid Robot ที่ได้รับทุนสนับสนุนจากศูนย์อิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (Nectec) เป็นงบประมาณทั้งสิ้นประมาณสามล้านบาท แผนภาพ ดังรูปที่ 4.1



รูปที่ 4.1 แสดงแบบจำลองโครงสร้างของหุ่นยนต์สองขาของ FIBO

หุ่นยนต์ต้นแบบที่สร้างขึ้นประกอบด้วย ส่วนของลำตัวและส่วนขา ดังรูปที่ 4.1 ในขาแต่ละข้างถูกออกแบบให้เป็นลักษณะข้อต่อหมุน (Revolute joint) โดยองศาอิสระ จำนวนองศาอิสระอย่างน้อยขาละ 6 องศาอิสระ เพื่อให้หุ่นยนต์สามารถเดินได้บนระนาบปริภูมิได้ คือข้อเท้า สององศาอิสระ หัวเข่าหนึ่งองศาอิสระ และสะโพก สามองศาอิสระ และได้ออกแบบระบบต้นกำลังโดยเลือกใช้มอเตอร์ไฟฟ้ากระแสตรงแบบป้อนกลับ (DC Servo Motor) ส่งถ่ายแรงบิดผ่านเฟืองแบบฮาร์โมนิก (Harmonic Gear) เพื่อเพิ่มความสามารถในการรับภาระแรงบิด (Torque Load) และส่งถ่ายกำลังจากมอเตอร์ไปยังเฟืองแบบฮาร์โมนิก โดยใช้ระบบสายพาน และเลือกใช้มอเตอร์แบบแบน (Pan Cake) ทำให้ในแต่ละข้อต่อไม่มียื่นออกมานอกขาหุ่นยนต์ หุ่นยนต์ต้นแบบที่สร้างขึ้นประกอบด้วยส่วนของลำตัวและส่วนขา ดังรูปที่ 4.1 ในส่วน

ระบบควบคุมการเดินประกอบด้วย คอมพิวเตอร์ 733 MHz ทำหน้าที่กำเนิดสัญญาณอ้างอิงให้กับตัวควบคุมการเคลื่อนที่ของมอเตอร์ (Motion controller) ซึ่งทำการคำนวณแรงบิดส่งไปยังตัวขยายสัญญาณ (Servo amplifier) เพื่อนำไปขับมอเตอร์ และรับค่าตำแหน่งเชิงมุมจากการหมุนของมอเตอร์จากตัววัดตำแหน่งเชิงมุมแบบต่อ เนื่อง (Incremental encoders) โดยทุกส่วนถูกติดตั้งบนลำ ตัวหุ่นยนต์และใช้แหล่งพลังงานไฟฟ้ากระแสตรงขนาด 34 โวลต์ 24 โวลต์และ ± 18 โวลต์ จากภายนอกเพื่อป้อนให้กับมอเตอร์และอุปกรณ์ที่ใช้ในการควบคุม ดังรูปที่ 4.2

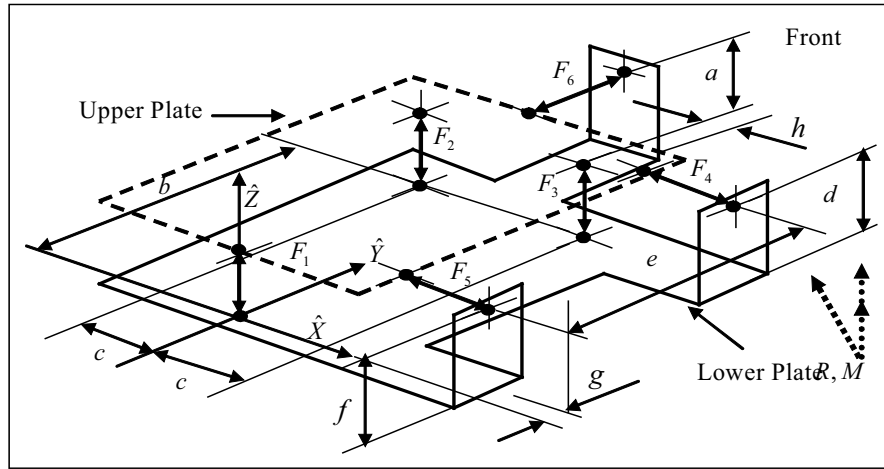


รูปที่ 4.2 หุ่นยนต์ฮิวแมนนอยด์ที่สร้างขึ้นที่ FIBO

จากการวิเคราะห์เสถียรภาพของหุ่นยนต์แบบ ZMP สามารถวิเคราะห์คำนวณจากแรงและโมเมนต์ ปฏิกริยาที่เกิดขึ้นที่เท้าทั้งสองได้ ดังนั้นการออกแบบหุ่นยนต์ ฮิวแมนนอยด์ของ FIBO ได้ทำการออกแบบอุปกรณ์วัดแรงซึ่งเป็นส่วนหนึ่งของเท้าหุ่นยนต์ อุปกรณ์วัดแรงที่ได้ออกแบบนี้สามารถวัดแรงทั้งหมดได้ในปริภูมิ กล่าวคือสามารถวัดแรงได้สามแกน และโมเมนต์ได้สามแกน โดยมีหลักการเบื้องต้นคือการออกแบบนี้ต้องมีการคำนวณทางคณิตศาสตร์ที่ไม่ซับซ้อน

อุปกรณ์วัดค่าแรงกระทำที่ฝ่าเท้าดังกล่าว ประกอบเข้าเป็นส่วนหนึ่งของเท้าและมีการปกปิดอย่างมิดชิด โดยอุปกรณ์ดังกล่าวนี้ประกอบด้วยตัวตรวจวัดแรง (Strain Gauge) แบบดึงและกดจำนวนหกตัว ยึดติดกับแผ่นรับน้ำหนักบนและล่างด้วยข้อต่อแบบทรงกลม (Ball Joint) การออกแบบได้ทำการวางตัวตรวจวัดแรง ให้มีความขนานและตั้งฉากกันเพื่อง่ายต่อการคำนวณ โดยมีแกนอ้างอิง X, Y และ Z โดยกำหนดพิกัดจากอ้างอิงอยู่ในตำแหน่งเดียวกับอุปกรณ์วัดแรงตัวที่หนึ่ง F_1 บนระนาบพื้นผิวสัมผัสของเท้า ซึ่งจุดที่แรงรวมทั้งหมดที่กระทำอยู่ในบริเวณพื้นผิวของฝ่าเท้า จุด ณ ตำแหน่งนั้นจะมีแรงและมี

โมเมนต์ทั้งสามแกน โดยที่ $F_1 - F_6$ คือแต่ตำแหน่งแรงที่จะทำการวัด $a-e$ คือระยะในการติดตั้งอุปกรณ์วัด และ R, M คือแรงปฏิกิริยาและโมเมนต์ปฏิกิริยา ซึ่งทั้งหมดมีความสัมพันธ์ดังรูปที่ 4.3



รูปที่ 4.3 แสดงแบบจำลองของอุปกรณ์วัดแรงที่เท้า

จากรูปสามารถสร้างสมการการถ่ายโอนแรงแบบตรงทั้งหมดหกสมการได้แก่สมการ R_x, R_y, R_z, x, y และ M_z สามารถเขียนให้อยู่ในรูปของการวัดแรงแบบหกแกนดังต่อไปนี้

$$R_x = F_4 + F_5 \quad (4.1)$$

$$R_y = -F_6 \quad (4.2)$$

$$R_z = (F_1 + F_2 + F_3) \quad (4.3)$$

$$M_z = F_6 \frac{(F_4 d + F_5 f - (F_2 - F_3) c)}{(F_1 + F_2 + F_3)} + \frac{(F_4 + F_5)(F_6 a + \frac{(F_2 + F_3) b}{(F_1 + F_2 + F_3)} + F_4(e + g) + F_5 g - F_6 h)}{(4.4)$$

และสามารถหาตำแหน่งของแรงลัพธ์ที่เทียบกับจุดอ้างอิงได้ดังสมการ

$$x = \frac{(F_4 d + F_5 f - (F_2 - F_3) c)}{(F_1 + F_2 + F_3)} \quad (4.5)$$

และ

$$y = \frac{(F_6 a + (F_2 + F_3) b)}{(F_1 + F_2 + F_3)} \quad (4.6)$$

และโมเมนต์รอบแกนอ้างอิงดังนี้ในแกน $\hat{X}$ และ $\hat{Y}$ ได้ดังนี้

$$M_x = R_z \times y \quad (4.7)$$

$$M_y = R_z \times x \quad (4.8)$$

ซึ่งได้รูปแบบทางคณิตศาสตร์ในการวัดแรงแบบหกแกนแบบสมบูรณ์ที่ได้อ้างอิงจากพิกัดจากอ้างอิงจากแบบจำลองข้างต้น และทำหุ่นยนต์ที่ได้จากการออกแบบนำไปผลิตขึ้นส่วนทำการติดตั้งอุปกรณ์วัดแรงทั้งหมดเข้าไปแสดงดังรูปที่ 4.4