บทที่ 2

ทฤษฎี และหลักการของระบบผู้เชี่ยวชาญ

2.1 คำจำกัดความของระบบผู้เชี่ยวชาญ

ระบบผู้เชี่ยวชาญ คือระบบของโปรแกรมคอมพิวเตอร์ที่เก็บรวบรวมฐานความรู้ที่ได้จากผู้เชี่ยวชาญมาใส่ให้เป็นฐานความรู้ให้กับระบบและมีกระบวนการอนุมานเพื่อที่จะนำไปสู่เป้าหมายหรือเป้าประสงค์คำตอบของปัญหานั้นๆ โดยกรรมวิธีในการนำไปสู่คำตอบเราเปรียบให้เป็นเสมือนกับการทำงานของระบบผู้เชี่ยวชาญนั้นเอง

ดังนั้นในการที่จะสร้างระบบผู้เชี่ยวชาญขึ้นมาใช้สำหรับในการแก้ปัญหาที่สลับซับซ้อนนั้น สิ่งที่ผู้สร้างจะต้องเข้าใจคือกรรมวิธีการทำงานของระบบผู้เชี่ยวชาญเป็นอย่างดี ซึ่งนั้นก็คือจะเป็นผลดีในการที่จะเก็บรวบรวมเอาความรู้ที่จะนำมาใช้ในกระบวนการอนุมานได้อย่างถูกต้องและสมบูรณ์ที่สุด

ดังนั้นเราพอที่จะสรุปได้ว่าระบบผู้เชี่ยวชาญเป็นโปรแกรมคอมพิวเตอร์ที่ใช้แก้ปัญหาโดยอาศัยความรู้ในสาขาใด สาขาหนึ่ง ที่รวบรวมความรู้จากผู้เชี่ยวชาญในสาขานั้นๆ และเก็บไว้เป็นฐานความรู้ในคอมพิวเตอร์โปรแกรมดังกล่าวต้องเป็นโปรแกรมเพื่อแก้ปัญหาโดยเฉพาะ ความสามารถในการแก้ปัญหาโปรแกรมนี้จะเทียบเท่าความสามารถของผู้มีความชำนาญระดับผู้เชี่ยวชาญนั้นๆ

2.2 องค์ประกอบของระบบผู้เชี่ยวชาญ

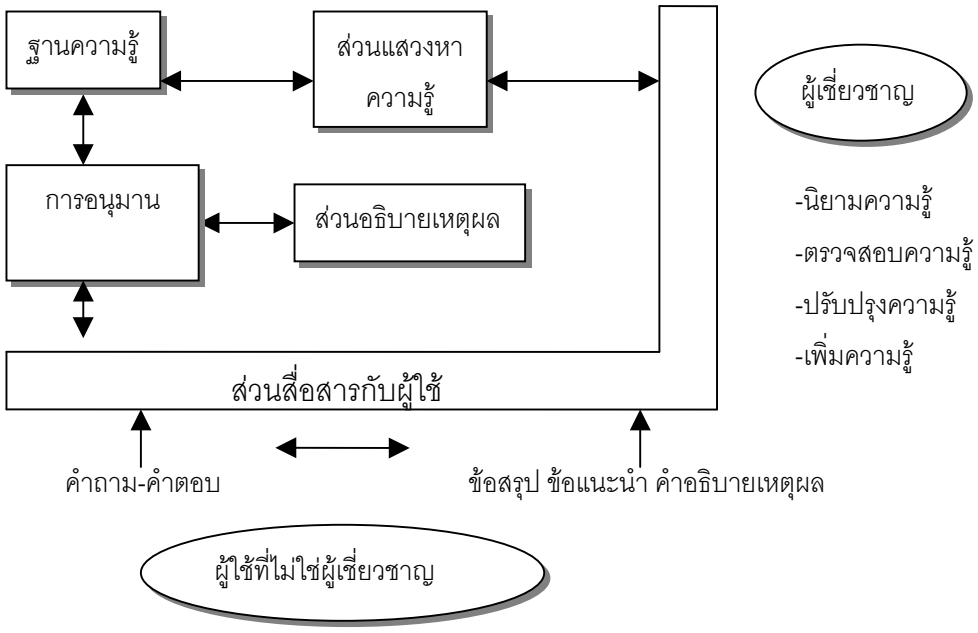
ระบบผู้เชี่ยวชาญโดยทั่วไปประกอบด้วยส่วนประกอบพื้นฐาน 5 ส่วน ดังภาพประกอบ 2-1 ส่วนที่เป็นหัวใจที่ขาดเสียมิได้ คือ ฐานความรู้ และเครื่องอนุมาน รายละเอียดโดยย่อของแต่ละส่วนสามารถอธิบายได้ดังนี้

1. ฐานความรู้ (Knowledge base) เป็นส่วนที่เก็บความรู้ในสาขาต่างๆ ที่ต้องการแก้ปัญหาเปรียบเสมือนฐานข้อมูล (database) ในระบบสารสนเทศ (Information System) ซึ่งเก็บในรูปที่คอมพิวเตอร์สามารถประมวณผลได้และมีการบำรุงรักษาฐานความรู้ให้ทันสมัยอยู่เสมอ ปัญหาหลักของฐานความรู้คือ การเลือกวิธีแสดงความรู้หรือโครงสร้างสำหรับเก็บข้อมูลที่เหมาะสมเพื่อไม่ให้เปลืองเนื้อที่ ขณะเดียวกันก็อยู่ในรูปที่วิเคราะห์หาเหตุผลจากความรู้นั้นออกมาได้ วิธีการแสดงความรู้ต่างๆเก็บไว้ในฐานความรู้ทำได้หลายแบบ เช่น แสดงความรู้ในรูปของกฎ

แสดงความรู้โดยใช้กรอบ แสดงความรู้โดยใช้ข่ายความหมาย เป็นต้น การจะเลือกใช้วิธีการใดขึ้นอยู่กับลักษณะของปัญหาที่จะศึกษาเป็นสำคัญ ระบบผู้เชี่ยวชาญบางระบบอาจใช้การแสดงความรู้หลายวิธีประกอบกัน

วิธีการแสดงความรู้ที่ดี จะต้องมีคุณสมบัติดังนี้

- มีสมรรถภาพในการแสดงความรู้ชนิดต่างๆได้ คือ ต้องสามารถบันทึกความรู้ ทั้งที่มีโครงสร้างความรู้ที่มีความไม่แน่นอน และความรู้ meta knowledge (ความรู้ที่เกี่ยวกับการใช้ความรู้อื่น) เป็นต้น โดยการใช้โครงสร้างที่สร้างง่าย แต่มีสมรรถภาพในการแสดงความรู้สูง
- มี modularity คือ ความสามารถในการแยกออกเป็นหัวข้อย่อย (module) เพื่อสามารถเพิ่มหรือแก้ไขฐานความรู้ ซึ่งจะทำให้เกิดความหยืดหยุ่นในการใช้ความรู้
- ง่ายต่อการจัดการเป็นคุณสมบัติที่ช่วยในการตรวจสอบฐานความรู้ เช่นช่วยในการตรวจดูความขัดแย้งในความรู้ การซ้ำกัน หรือความผิดพลาดในความรู้
- ง่ายต่อการเข้าใจของมนุษย์ ช่วยทำให้การสร้างส่วนอธิบายในระบบผู้เชี่ยวชาญง่ายขึ้นและช่วยในตรวจความผิดพลาดในการป้อนความรู้เข้าไปในฐานความรู้ด้วย
- เข้ากันได้ดีกับการอนุมานเนื่องจากการอนุมานต้องใช้ความรู้ในฐานความรู้เป็นข้อมูล



ภาพประกอบ 2-1 องค์ประกอบของระบบผู้เชี่ยวชาญ

2. การอนุมาน (Inference engine) เป็นส่วนของโปรแกรมที่ใช้ในการคิดหาเหตุผลโดยอาศัยความรู้จากฐานความรู้ข้างต้น เพื่อให้ได้วิธีแก้ปัญหาออกมา วิธีการหาเหตุผลปกติจะสัมพันธ์กับลักษณะการแทนความรู้ที่เลือกใช้ในการสร้างฐานข้อมูล

3. ส่วนการสื่อสารกับผู้ใช้ (User Interface module) เป็นส่วนของโปรแกรมที่ใช้สื่อสารระหว่างผู้ใช้กับระบบ เช่น อาจเป็นการตั้งคำถามมายังผู้ใช้เพื่อให้ผู้ใช้ป้อนข้อมูลที่จำเป็นเพื่อนำไปใช้ในการแก้ปัญหาต่อไป หรือผู้ใช้อาจตั้งคำถามให้ระบบอธิบายว่าระบบกำลังทำอะไรอยู่หรือระบบช่วยหาคำตอบให้ ถ้าการสื่อสารระหว่างผู้ใช้กับระบบผู้เชี่ยวชาญทำได้ง่ายระบบผู้เชี่ยวชาญนั้นก็จะมีประโยชน์ต่อผู้ใช้มากขึ้น

4. ส่วนแสวงหาความรู้ (Knowledge acquisition module) ในการนำความรู้จากผู้เชี่ยวชาญมาสร้างความรู้จำเป็นต้องมีการรวบรวมความรู้ของผู้เชี่ยวชาญแทนความรู้นั้นให้อยู่ในรูปที่คอมพิวเตอร์สามารถประมวลผล คิดค้นหาเหตุผลหรือคำตอบจากความรู้นั้นได้ และรวบรวมให้เป็นฐานความรู้ขึ้นมา กระบวนการเหล่านี้เรียกว่ากระบวนการแสวงหาความรู้ (Knowledge acquisition) ดังนั้นโปรแกรมส่วนนี้ก็คือส่วนที่ช่วยผู้ชำนาญการในการนิยามฐานความรู้ขึ้นมารวมทั้งช่วยในการปรับปรุงฐานความรู้นั้นให้ทันสมัยอยู่เสมอ

5. ส่วนอธิบายเหตุผล (Explanation module) คือ เมื่อผู้ใช้ปรึกษาระบบชำนาญการและคำตอบออกมาเพียงคำตอบเดียว ผู้ใช้อาจไม่มั่นใจว่าคำตอบที่ได้นั้นจะนำไปสู่การแก้ไขปัญหาได้จริง การอธิบายเหตุผลให้ผู้ใช้ทราบอาจทำให้ผู้ใช้นำคำตอบที่ได้ไปใช้งานอย่างมั่นใจโปรแกรมส่วนนี้จึงเป็นส่วนที่จำเป็นที่ใช้อธิบายเหตุผลให้ผู้ใช้ได้ทราบว่าข้อสรุปของระบบที่เป็นคำตอบออกมาได้อย่างไร

ในระบบผู้เชี่ยวชาญบางระบบจะไม่มีส่วนประกอบครบทั้ง 5 ส่วนดังกล่าวข้างต้นก็ได้ แต่ที่ขาดไม่ได้แน่ๆ คือ ส่วนของฐานความรู้และเครื่องอนุมาน ซึ่งจะกล่าวในรายละเอียดต่อไป

2.2.1 **ฐานความรู้**

ฐานความรู้เป็นส่วนที่เก็บความรู้ในสาขาที่ต้องการแก้ไขไว้ในรูปแบบที่คอมพิวเตอร์สามารถประมวลผลได้ และมีการบำรุงรักษาฐานความรู้ให้ทันสมัยอยู่เสมอ โดยปกติแล้วความรู้ที่รวบรวมมามักจะมีมากมาย ปัญหาที่สำคัญประการหนึ่งในการสร้างฐานความขึ้นมารู้คือทำอย่างไรจึงจะแทนความรู้จำนวนมากเหล่านั้น เพื่อเก็บไว้ในคอมพิวเตอร์ได้โดยไม่เปลืองเนื้อที่มากนัก ในขณะเดียวกันก็อยู่ในรูปที่วิเคราะห์หาเหตุผลจากความรู้นั้นออกมาได้ โดยรูปแบบของกฎจะมีลักษณะเป็นลิสต์ซึ่งประกอบด้วยชื่อของกฎ เงื่อนไข และผลสรุป

ในระบบผู้เชี่ยวชาญจำเป็นต้องมีฐานความรู้ที่จะต้องนำมาสำหรับการอนุมานในระบบ ซึ่งฐานความรู้เป็นส่วนที่ใช้เก็บความรู้ต่างๆ ทุกประเภทที่เกี่ยวข้องกับปัญหาไม่ว่าจะได้มาจากตำราหรือมาจากประสบการณ์ของผู้ชำนาญการเอง ซึ่งมีการแบ่งฐานความรู้ออกเป็น 2 ชนิดใหญ่ คือ ฐานความจริง และฐานกฎ ซึ่งสามารถที่จะอธิบายได้ดังต่อไปนี้ คือ

1. ความรู้ที่อยู่ในลักษณะของความจริง (Fact) หรือ สิ่งที่ปฏิเสธไม่ได้ คือความรู้ที่ได้มาจากตำราหรือได้จากประสบการณ์ของผู้ชำนาญการ

2. ความรู้ที่อยู่ในลักษณะของกฎ (Rule) คือการแสดงความรู้ในรูปของกลุ่มความจริง ส่วนที่บอกถึงขั้นตอน หรือการปฏิบัติในการแก้ไขปัญหา

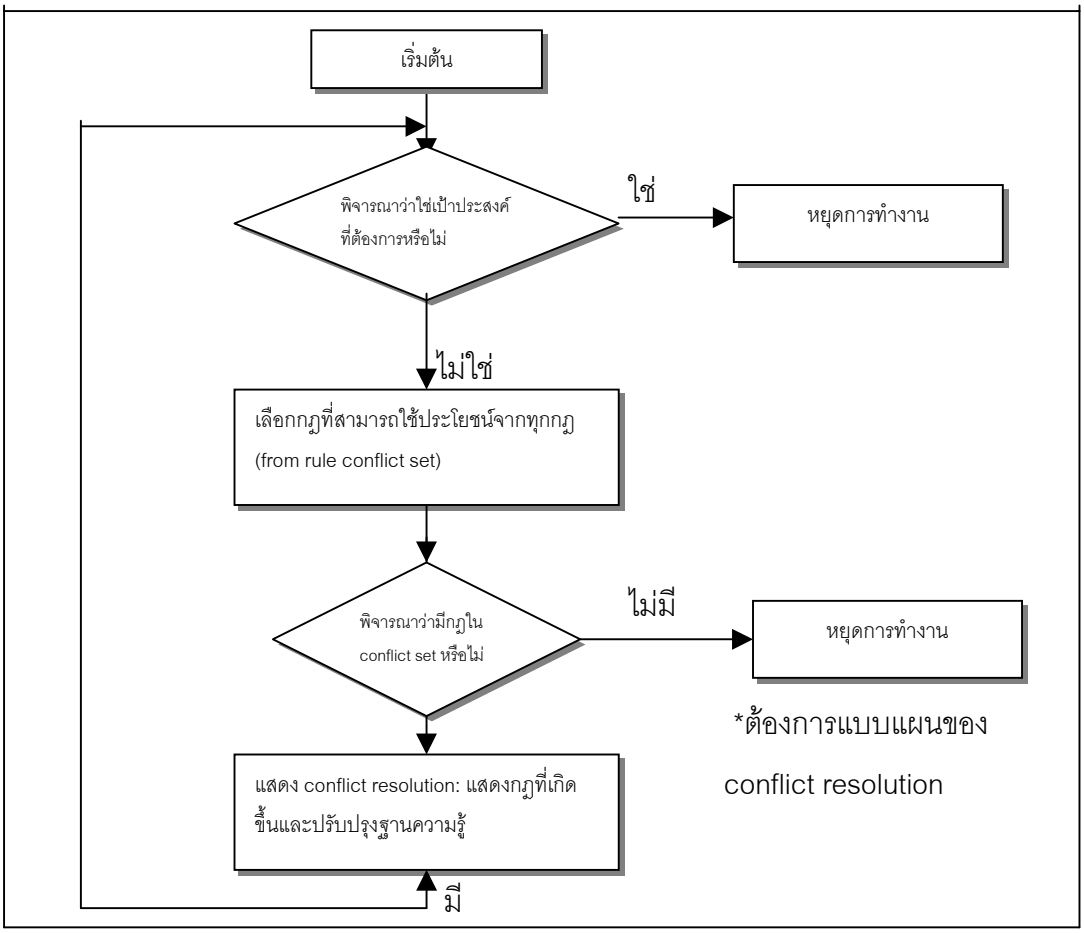
2.2.2 การอนุมาน

การอนุมานเป็นส่วนที่สำคัญที่สุดเมื่อเราทำการออกแบบฐานความรู้เสร็จสิ้นเรียบร้อยแล้ว ส่วนนี้เป็นส่วนหนึ่งของระบบเปลือกผู้เชี่ยวชาญ ซึ่งการอนุมานจะเป็นตัวกำหนดความสามารถของระบบ กำหนดความเร็ว และความถูกต้องของการแก้ปัญหา โดยหน้าที่หลักของการอนุมานคือเป็นตัวดึงความรู้จากฐานความรู้มาใช้ในการให้คำปรึกษาหรือแนะนำต่อผู้ใช้ โดยการดำเนินการนำเอาเงื่อนไขของแต่ละกฎมาถามต่อผู้ใช้ เพื่อพิสูจน์ค่าความจริงของแต่ละเงื่อนไข ในกฎเพื่อดึงเอาข้อสรุปที่ได้มาใช้เป็นคำตอบของผู้เชี่ยวชาญ พร้อมแสดงถึงความน่าจะเป็นของข้อสรุปนั้นๆ ด้วย

โดยการอนุมานที่ใช้ในระบบผู้เชี่ยวชาญนั้นจะมีอยู่ด้วยกันหลายวิธีขึ้นอยู่กับลักษณะของปัญหาและรูปแบบของการแทนความรู้ เช่น ในระบบผู้เชี่ยวชาญที่มีการทำงานเป็นลำดับขั้นตอนและใช้การแทนความรู้ในรูปของกฎ จะมีวิธีการอนุมานอยู่ด้วยกัน 3 วิธี คือ การอนุมานแบบเดินหน้า (Forward Chaining inference) การอนุมานแบบถอยหลัง (Backward Chaining Inference) และการอนุมานแบบผสม (Mix Chaining Inference) ซึ่งสามารถอธิบายหลักการของการอนุมานแต่ละอย่างได้ดังต่อไปนี้ คือ

การอนุมานแบบเดินหน้า

การอนุมานจะเริ่มต้นจากข้อมูลความจริงที่มีอยู่ในฐานข้อมูล โดยจะถูกเปรียบเทียบกับเงื่อนไขของกฎ ซึ่งกฎที่เงื่อนไขตรงกับข้อเท็จจริงที่มีอยู่ก็จะถูกปฏิบัติการตามข้อสรุปในส่วนของ THEN ซึ่งกฎต่างๆ จะถูกอนุมานในลักษณะนี้ไปเรื่อยๆ จนกว่าจะได้คำตอบหรือบรรลุเป้าหมาย ซึ่งเป็นกระบวนการในการสร้างฐานความรู้ขึ้นมาใหม่



ภาพประกอบ 2-2 การทำงานของการอนุมานแบบเดินหน้า [ที่มา: Robert, 1990]

ข้อดีของการอนุมานแบบเดินหน้า คือ จะค้นหาทุกเป้าประสงค์ที่เป็นไปได้ โดยมีข้อเสียคือ ในบางครั้งเป้าประสงค์ที่ได้ไม่ใช่สิ่งที่ต้องการ ดังนั้นจึงเป็นการเสียเวลาอย่างมาก ต่อระบบที่ต้องค้นหาคำตอบของแต่ละปัญหา และเราสามารถแสดงการอนุมานแบบเดินหน้าได้ดังภาพประกอบ 2-2 ซึ่งแสดงการทำงานของการอนุมาน และภาพประกอบ 2-3 และ 2-4 เป็นตัวอย่างของการอนุมานแบบเดินหน้า กฎและความจริงที่กำหนดขึ้นจะเป็นส่วนประกอบของฐานความรู้เริ่มต้น และลำดับของการพิจารณาเหตุผลแบบเดินหน้า ตามลำดับ

กำหนดให้มีกฎและความจริงเริ่มต้นคือ

กฎที่1. P_1 --------> P_2

กฎที่2. P_2 --------> P_3

กฎที่3. P_3 --------> P_4

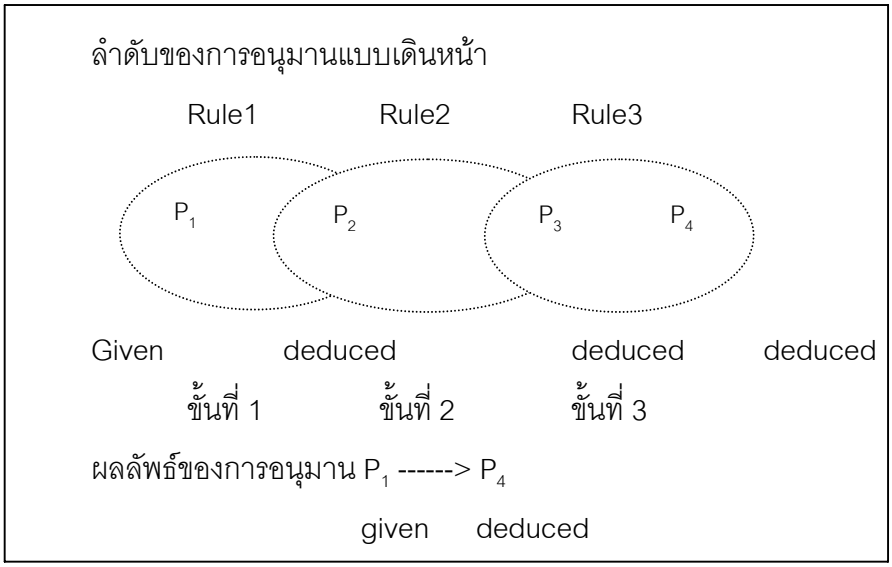
ความจริงเริ่มต้น P_1 เป็นจริง

ขั้นที่ 1 ใช้กฎที่ 1. $[(P_1 \text{ -----> } P_2) \wedge (P_1 \text{ เป็นจริง})] \text{ --------> } (P_2 \text{ เป็นจริง})$

ขั้นที่ 2 ใช้กฎที่ 2. $[(P_2 \text{ -----> } P_3) \wedge (P_2 \text{ เป็นจริง})] \text{ --------> } (P_3 \text{ เป็นจริง})$

ขั้นที่ 3 ใช้กฎที่ 3. $[(P_3 \text{ -----> } P_4) \wedge (P_3 \text{ เป็นจริง})] \text{ --------> } (P_4 \text{ เป็นจริง})$

ภาพประกอบ 2-3 ลำดับการพิจารณาเหตุผลของการอนุมานแบบเดินหน้า [ที่มา: Robert, 1990)

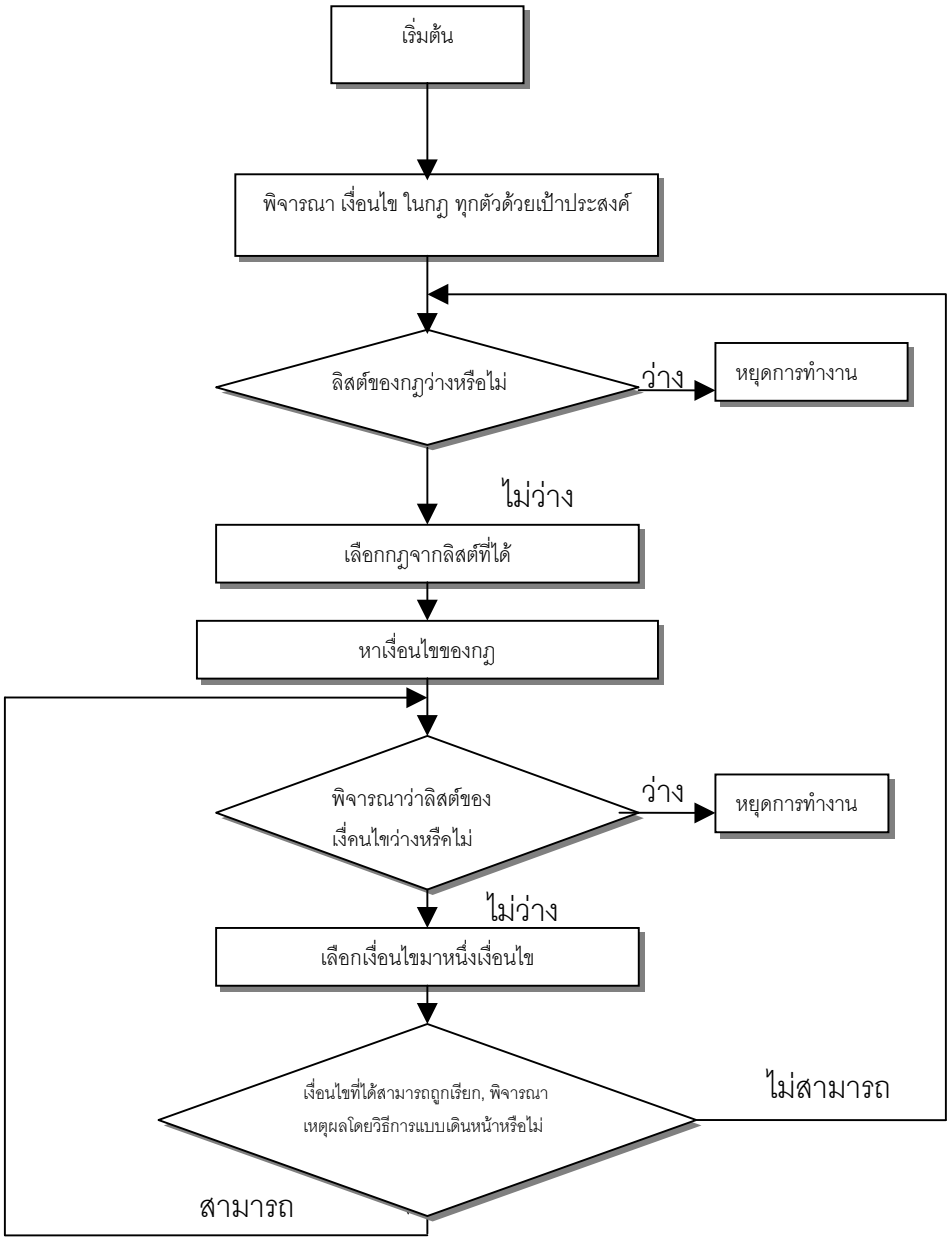


ภาพประกอบ 2-4 ลำดับการพิจารณาผลของการอนุมานแบบเดินหน้า[ที่มา: Robert, 1990]

การอนุมานแบบถอยหลัง

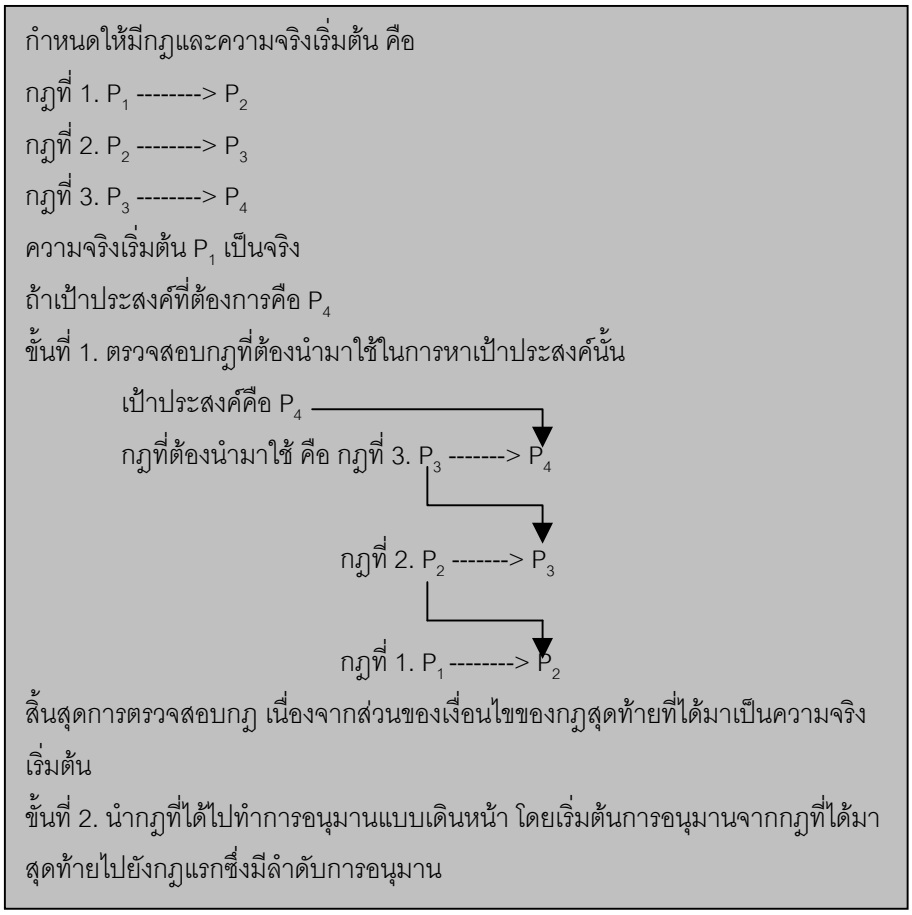
การอนุมานแบบถอยหลัง จะเริ่มต้นจากเป้าหมาย ซึ่งเป็นสมมติฐานที่สนใจไปสู่กฎที่จะนำให้เป้าหมายบรรลุผล ซึ่งกฎดังกล่าวจะต้องมีข้อสรุปเหมือนกับเป้าหมาย หลังจากนั้นเงื่อนไขของกฎ

นี้จะต้องถูกตรวจสอบว่าเป็นจริงตามข้อเท็จจริงที่มีอยู่ในฐานข้อมูลหรือไม่ ถ้าทุกเงื่อนไขของกฎนี้เป็นจริง เป้าหมายดังกล่าวก็บรรลุผลเป็นจริง แต่ถ้ามีบางเงื่อนไขหรือทุกเงื่อนไขของกฎนี้ไม่เป็นจริงตามข้อเท็จจริงในฐานข้อมูล เงื่อนไขที่ไม่เป็นจริงเหล่านี้ก็จะกลายเป็นเป้าหมายย่อยๆ เพื่อการอนุมานในขั้นต่อๆ ไป ซึ่งจะดำเนินไปในแบบลูกโซ่ จนกว่าจะพบกฎที่มีเงื่อนไขจากข้อเท็จจริง และทำให้เป้าหมายย่อยเหล่านั้นบรรลุผล ซึ่งก็เป็นการพิสูจน์สมมติฐานดังภาพประกอบ 2-5



ภาพประกอบ 2-5 การทำงานของการอนุมานแบบถอยหลัง [ที่มา: Robert, 1990]

ข้อดีของการอนุมานแบบถอยหลัง คือ จะค้นหาเฉพาะเป้าประสงค์ที่ต้องการเท่านั้นจึงใช้เวลาน้อย แต่ก็มีข้อเสียคือ ในบางครั้งอาจจะหาเป้าหมายไม่ได้เลย เนื่องจากเงื่อนไขในการที่จะนำไปสู่เป้าประสงค์นั้นไม่ถูกต้อง ตัวอย่างดังภาพประกอบ 2-6



ภาพประกอบ 2-6 ตัวอย่างการอนุมานแบบถอยหลัง [ที่มา: Robert, 1990]

การอนุมานแบบผสม

ลักษณะของการอนุมานแบบผสม จะเป็นลักษณะของการอนุมานที่รวมกันระหว่างการอนุมานแบบเดินหน้าและถอยหลัง กล่าวคือ เริ่มต้นระบบจะรับเป้าประสงค์จากผู้ใช้เหมือนกับการอนุมานแบบถอยหลัง จากนั้นจะนำเป้าหมายที่รับนั้นไปทำการอนุมานแบบถอยหลัง ซึ่งถ้าเป้าประสงค์ดังกล่าวเป็นจริง ระบบก็จะนำเส้นทางของการอนุมานนั้นไปทำการอนุมานแบบเดินหน้าเพื่อให้ได้ความจริงใหม่ ตามเส้นทางของการอนุมานแบบถอยหลัง

2.3 ภาษา LISP

ภาษา LISP เป็นภาษาทางคอมพิวเตอร์ชั้นสูงภาษาหนึ่ง เป็นภาษาที่ใช้ได้ดีในการประมวลผลทางสัญลักษณ์ (Symbol) จึงเป็นภาษาที่ใช้กันอย่างแพร่หลายในทางด้านปัญญาประดิษฐ์ (Artificial Intelligence :AI) โดยลักษณะพิเศษของภาษานี้คือ เป็นภาษาที่มีการตอบรับทันทีที่โปรแกรมถูกป้อนเข้าไป และจะมีการตรวจสอบไวยากรณ์ของภาษาด้วยตัวมันเองโดยที่ผู้ใช้ไม่จำเป็นต้องออกจากสภาพแวดล้อมของภาษา LISP จะส่งผลลัพธ์ของการหาค่า (Evaluation) ทันทีที่มีการพิมพ์โปรแกรมสิ้นสุด ดังนั้นในการโปรแกรมของภาษา เราไม่จำเป็นต้องมีการประกาศชนิดของตัวแปรที่ถูกใช้ในโปรแกรม

เหตุผลในการเลือกใช้ภาษา LISP

- โครงสร้างทางข้อมูลของภาษา LISP คือ List ซึ่งเป็นโครงสร้างที่โปรแกรมปัญญาประดิษฐ์ส่วนมากใช้ในการแทนความรู้

- สามารถรวบรวมข้อเท็จจริงเกี่ยวกับสิ่งต่างๆ ได้ง่าย โดยการแทนข้อมูลต่างๆ อยู่ในรูป Property list

- โครงสร้างในการควบคุมภาษา LISP ที่สำคัญคือ Recursion ซึ่งเหมาะกับการแก้ปัญหาหลายแบบ

- การกำหนดค่าให้กับตัวแปรจะกำหนดช่วงไหนก็ได้

ในงานวิจัยนี้ใช้ภาษา LISP เป็นภาษาหลักในการเขียนโปรแกรมควบคุมส่วนต่างๆ เช่น การอนุมาน ฐานความรู้ หรือส่วนที่ใช้สำหรับการติดต่อกับผู้ใช้ ซึ่งภาษา LISP ที่ใช้นี้ทำงานบนระบบปฏิบัติการลีนุกซ์

2.4 ระบบปฏิบัติการลีนุกซ์ (Linux)

ลีนุกซ์เป็นระบบปฏิบัติการแบบ 32 บิตที่พัฒนาขึ้นมาจากระบบปฏิบัติการยูนิกซ์ (Unix) สำหรับเครื่องพีซี และแจกจ่ายให้ใช้ฟรี พร้อมทั้งยังสนับสนุนการใช้งานแบบหลายงานพร้อมๆกัน (Multitasking), หลายผู้ใช้พร้อมๆกัน (multi-user), มีระบบ X-Window ซึ่งเป็นระบบการติดต่อกับผู้ใช้แบบกราฟฟิกที่ไม่ขึ้นกับโอเอสหรือฮาร์ดแวร์ใดๆ (โดยมากมักใช้กันแพร่หลายอยู่ในระบบยูนิกซ์ตัวอื่น) และมาตรฐานการสื่อสาร TCP/IP ที่ใช้เป็นมาตรฐานการสื่อสารในอินเตอร์เนตมาให้ในตัว ในรุ่นล่าสุดยังสนับสนุนการทำงานแบบมัลติโปรเซสเซอร์ด้วย โดยความหมายทางเทคนิคแล้ว ลีนุกซ์เป็นเพียงแก่นหรือเคอร์แนล (Kernel) ของระบบปฏิบัติการ ซึ่งทำหน้าที่ในด้านของการ

จัดสรรทรัพยากรในระบบ การประมวณการจัดการไฟล์และอุปกรณ์ I/O ต่างๆ แต่ในทางตรงกันข้าม ผู้ใช้ทั่วไปจะรู้จักลีนุกซ์โดยผ่านทางแอปปลิเคชั่นและระบบอินเตอร์เฟซที่เขาเหล่านั้นเห็น (เช่น Shell หรือ X-Window) เพียงเท่านั้น

การพัฒนาระบบงานบนลีนุกซ์ ลีนุกซ์ได้เตรียมเครื่องมือพัฒนาโปรแกรมไว้อย่างครบครัน ซึ่งมีตั้งแต่แอปปลิเคชั่นมาตรฐานคือ C/C++ คอมไพเลอร์ของ GNU และหากต้องการพัฒนาระบบบน X-Window ก็มีเครื่องมือของ Tcl/Tk เตรียมไว้ให้ด้วย สำหรับคอมไพเลอร์อื่นๆ เช่น Perl, Smalltalk, Pascal, LISP เป็นต้น นอกจากนี้ยังมีฐานข้อมูลที่สามารถรองรับการเขียนโปรแกรมแบบ X-Base (dBase,FoxBASE) หรือ FoxPro ล่าสุดลีนุกซ์มีคอมไพเลอร์จาวาให้ด้วย

เหตุผลที่เลือกใช้ระบบปฏิบัติการลีนุกซ์

1. เป็นระบบปฏิบัติการที่เสถียร ไม่ค่อยมีการเปลี่ยนแปลงเมื่อเทียบกับระบบปฏิบัติการวินโดว์
2. เป็นระบบปฏิบัติการแบบเปิด บุคคลทั่วไปสามารถนำระบบไปพัฒนาต่อได้
3. เป็นระบบปฏิบัติการที่สามารถดาวน์โหลดได้ฟรีจากอินเตอร์เนต
4. เป็นระบบปฏิบัติการที่เหมาะสมในการสร้างระบบควบคุม
5. เป็นระบบปฏิบัติการที่ไม่จำเป็นต้องใช้ทรัพยากรทางคอมพิวเตอร์มากนัก
6. โปรแกรมระบบเปลือกผู้เชี่ยวชาญในส่วนโครงสร้างฐานความรู้ ส่วนการอนุมานความรู้ ส่วนติดต่อกับผู้ใช้ และส่วนติดต่อกับภายนอกที่จะนำมาใช้นั้น ทำงานบนระบบปฏิบัติการลีนุกซ์
7. เป็นระบบปฏิบัติการที่มีคอมไพเลอร์ภาษา LISP ซึ่งเป็นภาษาที่ใช้ในการพัฒนาโปรแกรมระบบเปลือกผู้เชี่ยวชาญ