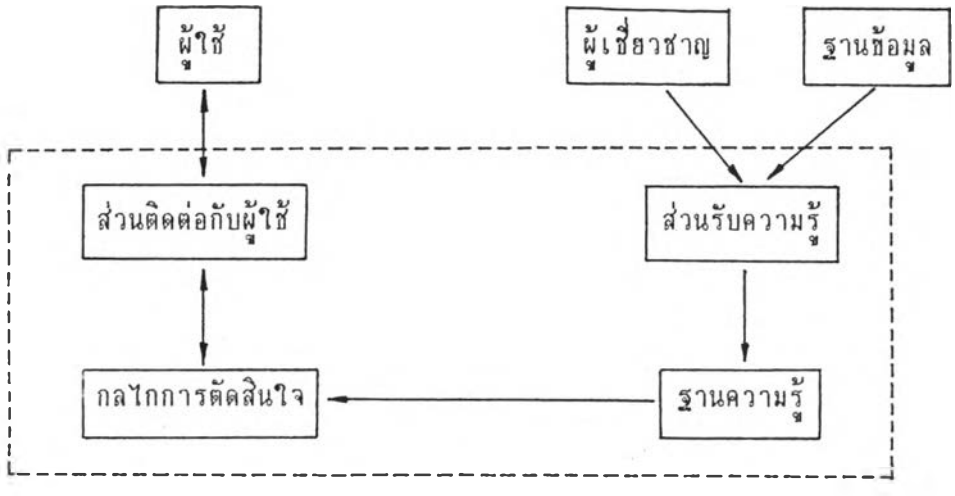
หอสมุดกลาง จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3

การออกแบบระบบเชี่ยวชาญเกี่ยวกับการวิเคราะห์สาเหตุขัดข้องของรถยนต์

ในบทนี้จะได้บรรยายถึงวิธีการ แนวทางแก้ปัญหา (approach) ตลอดจนการออกแบบระบบเชี่ยวชาญเกี่ยวกับการวิเคราะห์สาเหตุขัดข้องของรถยนต์ โดยจะกล่าวรายละเอียดในวิธีการของแต่ละขั้นตอน



รูปที่ 3.1 แสดงโครงสร้างของระบบเชี่ยวชาญ

ฐานความรู้ : ใช้วิธีการจัดเก็บความรู้แบบจัดเก็บเป็นกฎ (rule based knowledge representation)

กลไกการตัดสินใจ : ใช้การอนุมานแบบย้อนกลับ (backward chaining)

ส่วนรับความรู้ : เป็นโปรแกรมที่พัฒนาขึ้นเพื่อรวบรวมความรู้จากผู้เชี่ยวชาญหรือข้อมูลจากแหล่งอื่น เช่น หนังสือ, วารสาร โดยจัดเก็บเป็นรูปแบบไว้ในฐานความรู้

ส่วนติดต่อกับผู้ใช้ : มีการใช้หน้าต่างหลายอัน การใช้หน้าต่างซ้อนสลับ การใช้รายการ และการใช้กราฟิกส์หรือรูปภาพประกอบ เพื่อให้ผู้ใช้ๆได้ง่าย และลดอัตราการผิดพลาดระหว่างคนและระบบ

3.1 ลักษณะการวิเคราะห์สาเหตุขัดข้องของรถยนต์

การพิจารณาอาการขัดข้อง จะกระทำภายใต้ระบบหนึ่งๆ ซึ่งจะบอกถึงสาเหตุที่เป็นไปได้เพื่อไปตรวจสอบตามสาเหตุนั้น ตัวอย่างเช่น การวิเคราะห์ว่าแรงดันน้ำมันเครื่องมากเกินไป จะกระทำภายใต้ระบบหล่อลื่น (Martin, 1978) ดังแสดงในตารางที่ 3.1

LUBRICATION SYSTEM PROBLEM DIAGNOSIS	
PROBLEM	POSSIBLE CAUSES
EXCESSIVE OIL PRESSURE	1. Oil viscosity too high 2. Pressure relief valve spring under too much tension 3. Pressure relief valve stuck 4. Main oil line from pump clogged 5. Defective gauge

ตารางที่ 3.1 แสดงการวิเคราะห์สาเหตุขัดข้องของรถยนต์

จากตารางที่ 3.1 อธิบายได้ดังนี้
ปัญหาแรงดันน้ำมันเครื่องมากเกินไปซึ่งอยู่ภายใต้การวิเคราะห์ด้านระบบหล่อลื่นนั้นมีสาเหตุที่เป็นไปได้คือ

1. ความหนืดของน้ำมันมากเกินไป
2. สปริงลิ้นควบคุมความดันมีความตึงมากเกินไป
3. ลิ้นควบคุมความดันติดขัด
4. ท่อส่งน้ำมันจากปั๊มน้ำมันเครื่องอุดตัน
5. เกจวัดแรงดันน้ำมันขัดข้อง

ในเบื้องต้นจะขอนิยามความหมายของคำต่างๆดังนี้

1. ระบบเชี่ยวชาญ (expert system) : ชุดโปรแกรมคอมพิวเตอร์ที่ใช้ความรู้จากฐานความรู้ในการแก้ปัญหาในขอบเขตหนึ่งๆ

2. ฐานความรู้ (knowledge base) : ฐานข้อมูลที่ประกอบด้วยกฎต่างๆเกี่ยวกับความรู้ในเรื่องหนึ่งๆ

3. ข้อเท็จจริง (fact) : ข้อมูลจริงที่รวบรวมได้

4. การจัดเก็บความรู้ (knowledge representation) : วิธีการจัดเก็บข้อมูลโดยจัดเก็บเป็นโครงสร้างไว้ในฐานความรู้

5. แอททริบิวท์ (attribute) : วัตถุซึ่งประกอบด้วยคุณสมบัติเฉพาะอันหนึ่งที่จะอธิบายวัตถุนั้น

6. อนุประโยค (clause) : แอททริบิวท์และค่าของแอททริบิวท์ที่ประกอบกันเพื่อทำให้มีความหมายแก่การเข้าใจ

7. กฎ (rule) : การจัดเก็บความรู้ โดยประกอบด้วยอนุประโยคส่วนเงื่อนไขและอนุประโยคส่วนสรุป

8. กลไกการตัดสินใจ (inference engine) : ส่วนหนึ่งในระบบเชี่ยวชาญที่ใช้ข้อมูลในฐานความรู้เพื่อการวินิจฉัย

3.2 การเลือกวิธีการจัดเก็บความรู้

ใช้การจัดเก็บความรู้โดยการใช้กฎ (rule based knowledge representation) กฎจะอยู่ในรูป

IF condition THEN conclusion (action) เช่น

IF the pH of the spill is less than 6

THEN the spill material is an acid

หรือยังเขียนได้อีกรูปแบบหนึ่งได้คือ condition ⟶ conclusion (action)

ในระบบเชี่ยวชาญซึ่งใช้กฎเป็นพื้นฐานนั้น (rule based expert system) ความรู้จะถูกจัดเก็บโดยกลุ่มของกฎ ซึ่งกฎจะถูกทดสอบกับข้อเท็จจริงหรือความรู้ในเรื่องนั้นๆ ถ้าเงื่อนไขส่วน IF ของกฎตรงกับข้อเท็จจริง ส่วนข้างหลัง THEN ในกฎจะถูกปฏิบัติการ (execute) และหลังการปฏิบัติการ ข้อสรุปที่ได้อาจเปลี่ยนแปลงข้อเท็จจริงในฐานความรู้ได้ เช่นเกิดการเพิ่มข้อเท็จจริงใหม่เข้าไป เป็นต้น

3.2.1 การออกแบบการจัดเก็บความรู้

ได้ออกแบบให้มีการจัดเก็บเป็นกฎ โดยมีความสัมพันธ์ในการสร้างกฎดังนี้

rule เกิดจาก IF clause และ THEN clause

IF clause เกิดจาก clause 1 ∧ clause 2 ∧ clause 3 ...

THEN clause เกิดจาก clause

clause เกิดจาก attribute และ value

ตัวอย่าง เช่น

กำหนด attribute คือ oil pressure และมี value คือ low และ excessive

clause ที่สามารถสร้างได้คือ

oil pressure low หรือ

oil pressure excessive

ในทำนองเดียวกัน ถ้ากำหนด attribute คือ oil และมี value คือ light และ too viscous

clause ที่สามารถสร้างได้คือ

oil light หรือ

oil too viscous

ดังนั้น สามารถสร้างกฎได้คือ

IF oil too viscous

THEN oil pressure excessive

3.2.1.1 การพิจารณากรณีการวนซ้ำ (cyclic cases)

ได้มีการพิจารณาว่าการใส่กฎเข้าไปในฐานความรู้ (knowledge base) จะมีโอกาสทำให้เกิดการวนซ้ำ (cyclic) ขึ้นหรือไม่ ซึ่งเมื่อกรณีเช่นนี้เกิดขึ้นแล้ว จะทำให้ไม่สามารถหาคำตอบที่ต้องการได้ ตัวอย่าง เช่น

RULE NO. 1 :

IF clause 1 THEN clause 2

RULE NO. 2 :

IF clause 2 THEN clause 3

RULE NO. 3 :

IF clause 3 THEN clause 1

ถ้าพิจารณากฎข้อที่ 1 clause 2 จะเป็นจริงได้ก็ต่อเมื่อ clause 1 เป็นจริง
และจากกฎข้อที่ 3 clause 1 จะเป็นจริงได้ก็ต่อเมื่อ clause 3 เป็นจริง
และจากกฎข้อที่ 2 clause 3 จะเป็นจริงได้ก็ต่อเมื่อ clause 2 เป็นจริง
จะเห็นว่า จะไม่สามารถหาข้อสรุปได้ เพราะความสัมพันธ์ของทั้ง 3 กฎนั้น ก่อให้เกิดการวนซ้ำ

ปัญหาข้อนี้จะได้มีการตรวจสอบเมื่อมีการเพิ่มกฎเข้าสู่ฐานความรู้ และเมื่อมีการตรวจสอบพบว่ากฎใดกฎหนึ่งทำให้เกิดการวนซ้ำแล้ว จะกำจัดกฎทั้งกลุ่มนั้นทิ้ง

3.2.1.2 <u>การพิจารณากรณีความขัดแย้ง</u> (contradiction cases)

เป็นการพิจารณาว่า การใส่กฎใหม่เข้าไปในฐานความรู้ (knowledge base) จะทำให้เกิดการขัดแย้งกับกฎเดิมที่อยู่ในฐานความรู้หรือไม่ ซึ่งเมื่อเกิดขึ้นแล้ว จะทำให้ไม่สามารถสรุปหาคำตอบที่แน่นอนได้ ตัวอย่าง เช่น

RULE NO. 1 :

IF clause 1 THEN clause 2

RULE NO. 2 :

IF clause 1 THEN not clause 2

จะเห็นว่า ถ้าเดิมมีกฎข้อ 1 อยู่ในฐานความรู้อยู่แล้ว การใส่กฎข้อที่ 2 ลงไป จะทำให้ความขัดแย้งขึ้นกับกฎข้อ 1 เพราะจะเกิดลักษณะที่ เมื่อ clause 1 เป็นจริง clause 2 จะเป็นจริงและ ไม่จริง ในเวลาเดียวกัน ซึ่งเป็นไปไม่ได้

ปัญหาข้อนี้จะได้มีการตรวจสอบเมื่อมีการเพิ่มกฎเข้าสู่ฐานความรู้ และเมื่อมีการตรวจสอบพบว่ากฎใดกฎหนึ่งทำให้เกิดความขัดแย้งแล้ว จะกำจัดกฎทั้งกลุ่มนั้นทิ้ง

3.2.2 การออกแบบฐานความรู้

ในที่นี้ประกอบด้วย

3.2.2.1 ฐานข้อมูลสถิต (static database)

เป็นฐานข้อมูลที่ไม่มีการเปลี่ยนแปลงในระหว่างการใช้ระบบเชี่ยวชาญ นั่นคือ จะเก็บความรู้ซึ่งเป็นกฎต่างๆไว้ในฐานข้อมูลสถิต

3.2.2.2 ฐานข้อมูลพลวัตร (dynamic database)

เป็นฐานข้อมูลที่สร้างขึ้นในระหว่างการใช้ระบบเชี่ยวชาญ (เช่น ข้อมูลจากการตอบคำถามของผู้ใช้) โดยการจัดสรรความจำ (allocate memory) ให้แต่ละโหนด (node) เมื่อไม่ต้องการใช้โหนด (node) เหล่านี้แล้ว จะทำการคืนความจำ (free memory) แก่หน่วยความจำหลัก

3.3 การเลือกกลไกการตัดสินใจ

ใช้การอนุมานแบบย้อนกลับ (backward chaining) โดยมีหลักการว่า จะเริ่มต้นจากสิ่งที่ต้องการพิสูจน์ และปฏิบัติการต่อกฎซึ่งมีความสัมพันธ์ตรงกับสิ่งที่พิสูจน์นั้น เช่น

RULE NO. 1 :

IF clause 1 THEN clause 2

RULE NO. 2 :

IF clause 2 THEN clause 3

RULE NO. 3 :

IF clause 3 THEN clause 4

การจะพิจารณาว่า clause 4 เป็นจริงหรือไม่นั้น จะพิจารณา clause 3 ในส่วนเงื่อนไขในกฎข้อที่ 3 แต่เนื่องจาก clause 3 จะเป็นจริงได้ก็ต่อเมื่อ clause 2 ในกฎข้อที่ 2 เป็นจริง และ clause 2 จะเป็นจริงได้ก็ต่อเมื่อ clause 1 ในกฎข้อที่ 1 เป็นจริง
ดังนั้น ถ้า clause 1 ในกฎข้อที่ 1 เป็นจริง จะทำให้ clause 4 ในส่วนสรุปของกฎข้อที่ 3 เป็นจริง เมื่ออนุมานย้อนกลับ

3.3.1 การใช้แผนภาพต้นไม้ในการกำหนดเส้นทาง

จะขอยกตัวอย่างของกฎเพื่อประกอบการอธิบายดังนี้

RULE NO. 10

IF starter motor defective

THEN engine not start

RULE NO. 20

IF battery terminal loosen

THEN engine not start

RULE NO. 30

IF battery defective

THEN engine not start

RULE NO. 40

IF starter brush worn

THEN starter motor defective

RULE NO. 50

IF starter armature grounded

THEN starter motor defective

RULE NO. 60

IF response from engine none

AND headlight dim

AND voltage in cell less than 1.5 v.

THEN battery defective

RULE NO. 70

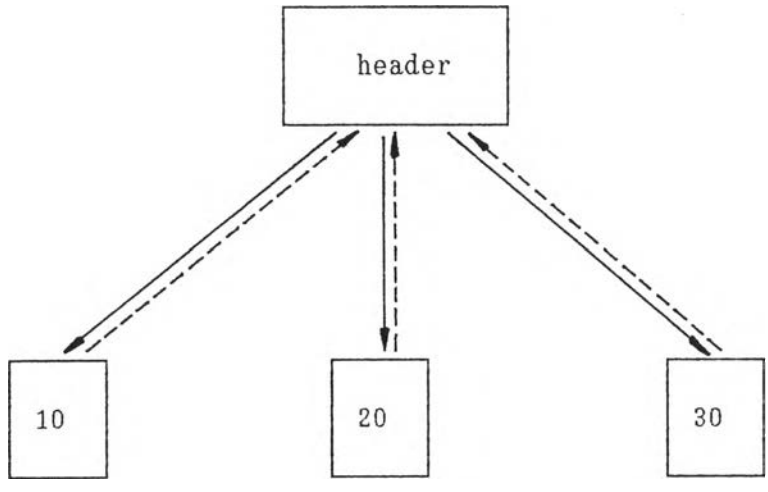
IF resistance reading for starter armature ground low

THEN starter armature grounded

017765

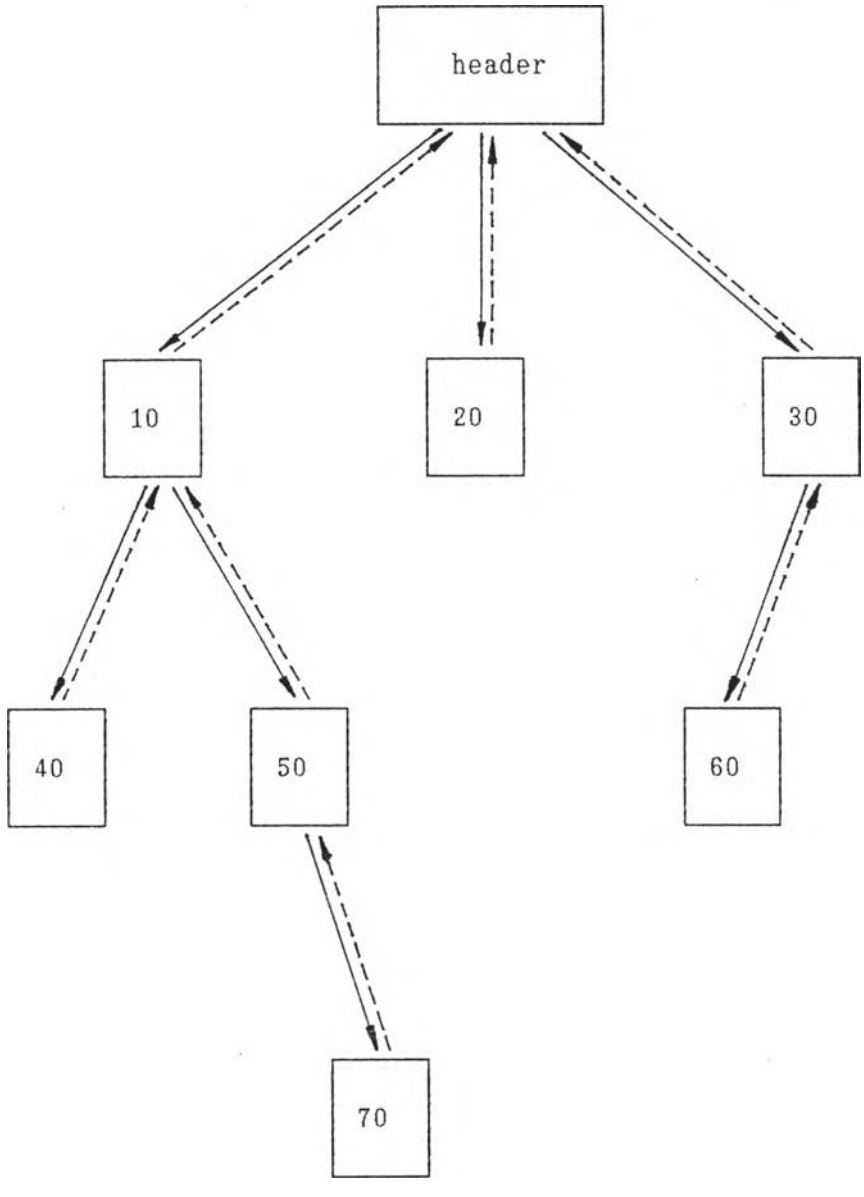
ในที่นี้ ส่วนหัว (header) เป็นอาการ (symptom) ที่เกิดขึ้นในระบบที่เลือก เช่น อาการ engine not start

ส่วนหัว (header) จะชี้ไปยังโหนดลูก (child node) ซึ่งก็คือกฎต่างๆที่ทำให้เกิดอาการนี้ เช่น



รูปที่ 3.2 แสดงเส้นทางจากส่วนหัวไปยังโหนดที่เกี่ยวข้อง

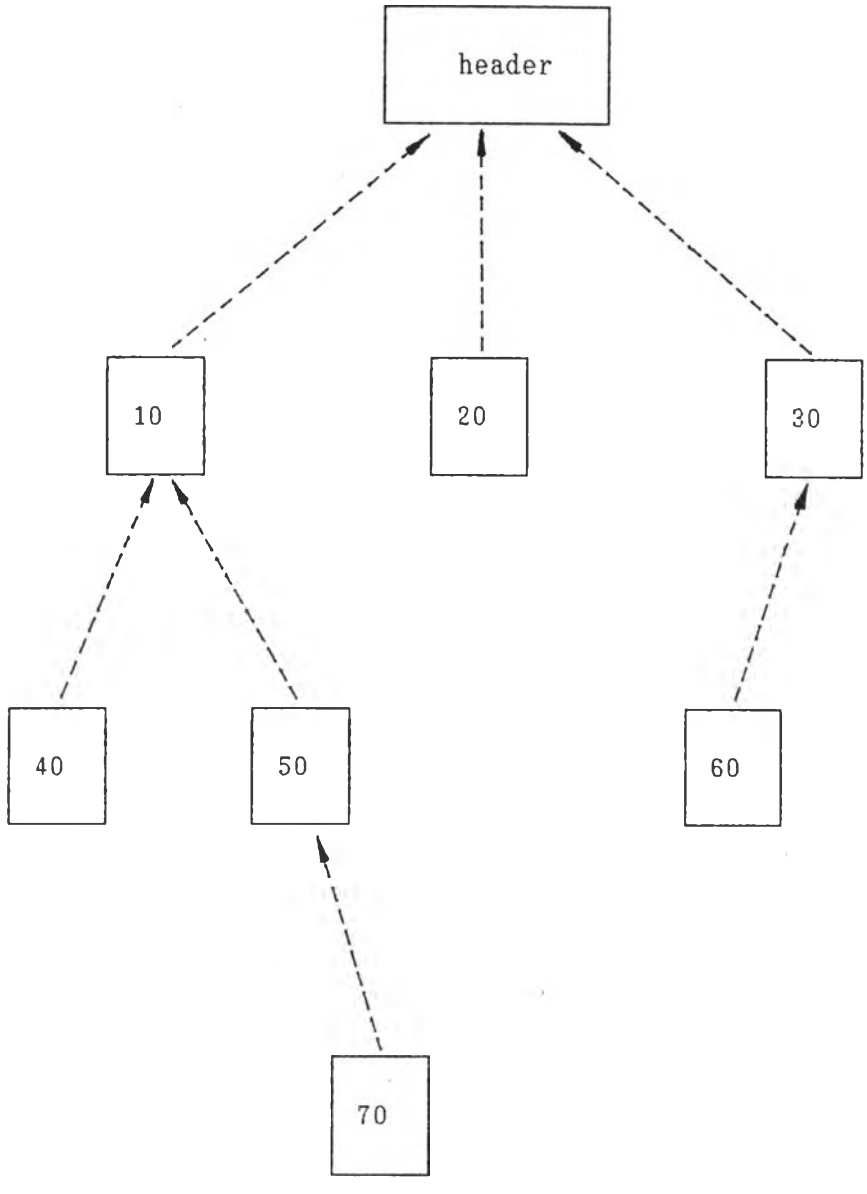
กฎต่างๆที่อ้างถึงนี้ บางกฎก็ยังไม่สามารถสรุปข้อเท็จจริงได้ เพราะอาจจะมีการอ้างไปถึงกฎอื่นอีก ดังนั้น จึงเกิดโหนดลูกขึ้น ขณะเดียวกัน โหนดลูก ก็จะมีตัวชี้กลับมายังโหนดพ่อแม่ด้วย เช่น



รูปที่ 3.3 แสดงเส้นทางจากโหนดพ่อแม่ไปยังโหนดลูก
เมื่อกฎพ่อแม่มีการอ้างถึงกฎที่เป็นโหนดลูก

3.3.2 การค้นหาแบบหยั่งรู้ (heuristic search)

เป็นวิธีการที่มุ่งหาคำตอบได้อย่างรวดเร็ว ตัวอย่าง เช่น เมื่อมีการกำหนดเส้นทางด้วยแผนภาพต้นไม้แล้ว การพิจารณาย้อนกลับไปว่ากฎใดจะโยงไปถึงส่วนหัวได้ ต้องพิจารณาจากโหนดใบ (leaf node) ฉะนั้นจึงนำกฎที่เป็นโหนดใบ (leaf node) ขึ้นมาพิจารณา โดยพิจารณาจากส่วนเงื่อนไขว่าเป็นจริงหรือไม่ ด้วยการตั้งคำถามเพื่อรับคำตอบในเชิงบวกหรือลบ (positive or negative) การค้นหาแบบหยั่งรู้จะเก็บข้อมูลของโหนดที่เป็นเชิงบวก (positive) ไว้ แล้วทำการอนุมานกลับไปยังโหนดพ่อแม่ เพื่อทำการวินิจฉัยส่วนเงื่อนไขในโหนดพ่อแม่อีก และจะทำแบบนี้ขึ้นไปเรื่อยๆจนกระทั่งถึงส่วนหัว (header) จึงจะสามารถสรุปคำตอบออกมาได้



รูปที่ 3.4 แสดงเส้นทางการอนุมานย้อนกลับจากโหนดใบไปยังส่วนหัว

3.4 ส่วนติดต่อกับผู้ใช้ (user interface)

เป็นส่วนที่ทำให้ผู้ใช้สามารถติดต่อกับระบบได้โดยไม่จำเป็นต้องรู้วิธีประมวลผลโดยละเอียด ส่วนนี้จะต้องทำให้ผู้ใช้มีความรู้สึกว่าการติดต่อกับระบบทำได้ง่ายและน่าใช้ มีประสิทธิภาพในการใช้และลดอัตราความผิดพลาดในระหว่างการติดต่อระหว่างคนและระบบ สำหรับการออกแบบในที่นี้จะสามารถให้ผู้ใช้เลือกตอบคำถามได้พร้อมกันหลายๆข้อ เพื่อให้ส่วนควบคุมนำคำตอบที่ได้รับไปพิจารณาเลือกเส้นทางค้นหาที่เป็นไปได้ และมีการใช้

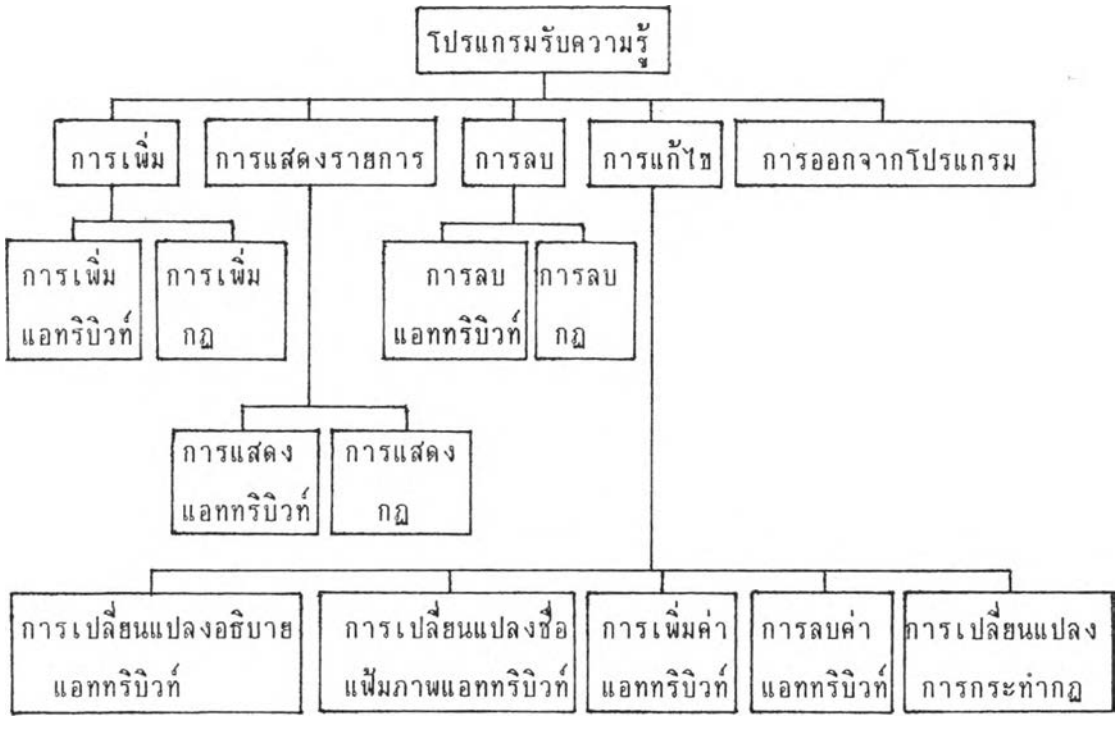
กราฟิกส์หรือรูปภาพประกอบ เช่น มีภาพประกอบซึ่งแทนกลุ่มของคำถามซึ่งจะให้ผู้ใช้เลือกตอบ ผู้ใช้อาจจะตอบหรือเลือกภาพอื่นซึ่งแทนกลุ่มคำถามอื่นต่อไปก็ได้ ตัวอย่าง เช่น ภาพจานจ่าย ซึ่งเมื่อแสดงขึ้นมาแล้ว ทำให้พอมองออกต่อไปว่า ควรจะเป็นคำถามที่เกี่ยวกับอุปกรณ์ของจานจ่าย เป็นต้น

3.5 การออกแบบการทำงานของโปรแกรม

จะแบ่งโปรแกรมหลักออกเป็น 2 ส่วน คือ

1. โปรแกรมรับความรู้
2. โปรแกรมการวินิจฉัย

3.5.1 โปรแกรมรับความรู้



รูปที่ 3.5 แสดงโปรแกรมรับความรู้

ทำหน้าที่เก็บรวบรวม แก้ไข หรือ ลบความรู้ซึ่งประกอบด้วยกฎเกณฑ์ต่างๆจากฐานความรู้ หน้าที่ต่างๆในโปรแกรมรับความรู้ประกอบด้วย

3.5.1.1 การเพิ่ม (add)

ประกอบด้วย

3.5.1.1.1 การเพิ่มแอททริบิวท์ (add attribute)

คือการเพิ่มแอททริบิวท์ และค่าของแอททริบิวท์ (value) ซึ่งอาจมีได้หลายๆค่าเข้าสู่ฐานความรู้ โปรแกรมจะตรวจสอบว่าแอททริบิวท์ใหม่ที่เพิ่มเข้าไปเกิดการซ้ำซ้อนกับแอททริบิวท์ เดิมหรือไม่ ถ้าเกิดการซ้ำ จะมีการแจ้งข้อผิดพลาดขึ้น

3.5.1.1.2 การเพิ่มกฎ (add rule)

เป็นการเพิ่มกฎซึ่งประกอบด้วย IF clause และ THEN clause เข้าสู่ฐานความรู้ โปรแกรมจะตรวจสอบดังนี้

1. เลขที่กฎไม่ซ้ำซ้อน
2. ทดสอบว่าเกิดการวนซ้ำ (cyclic cases) หรือไม่ ถ้าเกิดจะต้องกำจัดทิ้ง (reject) และแจ้งข้อผิดพลาด
3. ทดสอบว่าเกิดการขัดแย้ง (contradictory rule) หรือไม่ ถ้าเกิดจะต้องกำจัดทิ้ง (reject)

ปกติการเพิ่มกฎสามารถจัดทำเป็นกลุ่มของกฎได้ และเมื่อกฎใดกฎหนึ่งที่เพิ่มเข้าไปเกิดข้อผิดพลาดขึ้น ไม่ว่าจะเป็น เกิดการวนซ้ำ หรือ เกิดการขัดแย้ง จะต้องกำจัด (reject) กฎทั้งกลุ่มนั้นด้วย

3.5.1.2 การแสดงรายการ (list)

ประกอบด้วย

3.5.1.2.1 การแสดงรายการแอททริบิวท์ (list attribute)

เป็นการแสดงแอททริบิวท์และค่าต่างๆทั้งหมดของแอททริบิวท์ นอกจากนี้ ยังแสดงรายการคธิบายแอททริบิวท์ (attribute description) ซึ่ง

จะอธิบายตอนเลือกตอบ และการแสดงชื่อแฟ้มข้อมูลที่เก็บภาพของแอททริบิวท์นี้ด้วย
การแสดงรายการแอททริบิวท์ เกิดขึ้นเมื่อป้อนคีย์ คือ แอททริบิวท์ที่ต้องถูกแสดงรายการ ในกรณีที่ไม่มีแอททริบิวท์นั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น

3.5.1.2.2 การแสดงรายการกฎ (list rule)

เป็นการแสดงกฎ ซึ่งประกอบด้วย ส่วนเงื่อนไขทั้งหมด (IF clause) และส่วนสรุป (THEN clause) นอกจากนี้ ยังแสดงการกระทำ (action) ของกฎนี้ เมื่อกฎนี้ถูกปฏิบัติ (execute) ด้วย
การแสดงรายการกฎ เกิดขึ้นเมื่อป้อนเลขที่กฎที่ต้องการ ในกรณีที่ไม่มีหมายเลขกฎนั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น

3.5.1.3 การลบ (delete)

ประกอบด้วย

3.5.1.3.1 การลบแอททริบิวท์ (delete attribute)

เป็นการลบแอททริบิวท์ และค่าของแอททริบิวท์ออกจากฐานความรู้ ซึ่งกระทำโดยการป้อนคีย์ซึ่งเป็นแอททริบิวท์ที่ต้องการให้ถูกลบ ในกรณีที่ไม่มีแอททริบิวท์นั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น นอกจากนี้ การลบแอททริบิวท์จะเกิดขึ้นได้ก็ต่อเมื่อไม่มีกฎข้อใดใช้แอททริบิวท์นั้นอยู่ และการลบค่าของแอททริบิวท์จะเกิดขึ้นได้ก็ต่อเมื่อไม่มีแอททริบิวท์อื่นใดใช้ค่านั้นอยู่เช่นกัน

3.5.1.3.2 การลบกฎ (delete rule)

เป็นการลบกฎออกจากฐานความรู้ ซึ่งกระทำโดยป้อนเลขที่กฎที่ต้องการให้ถูกลบ ในกรณีที่ไม่มีกฎนั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น
การลบกฎ จะลบทั้งส่วนเงื่อนไข (IF clause) ทั้งหมด และส่วนสรุป (THEN clause) ของกฎ นอกจากนี้ ยังรวมถึงเลขที่กฎนั้นด้วย

3.5.1.4 การแก้ไข (update)

ประกอบด้วย

3.5.1.4.1 การเปลี่ยนแปลงการอธิบายแอททริบิวท์ (change attribute description)

เป็นการเปลี่ยนแปลงรายการอธิบายแอททริบิวท์ ซึ่งการอธิบายนี้จะใช้แสดงเมื่อมีการตั้งคำถามเกี่ยวกับแอททริบิวท์นั้น การเปลี่ยนแปลงกระทำได้โดยป้อนคีย์ของแอททริบิวท์ที่ต้องการเปลี่ยนแปลงรายการอธิบาย ในกรณีที่ไม่มีแอททริบิวท์นั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น

3.5.1.4.2 การเปลี่ยนแปลงชื่อแฟ้มภาพของแอททริบิวท์ (change attribute picture)

เป็นการเปลี่ยนชื่อแฟ้มที่เก็บรูปภาพของแอททริบิวท์ ซึ่งจะใช้เป็นส่วนแสดงภาพออกมาประกอบกับแอททริบิวท์เพื่อทำให้เข้าใจขึ้น การเปลี่ยนแปลงกระทำได้โดย ป้อนคีย์ของแอททริบิวท์ที่ต้องการเปลี่ยนชื่อแฟ้มที่เก็บรูปภาพ ในกรณีที่ไม่มีแอททริบิวท์นั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น

3.5.1.4.3 การเพิ่มค่าของแอททริบิวท์ (add attribute value)

การเพิ่มค่าของแอททริบิวท์ ต้องพิจารณาด้วยว่าค่าที่เพิ่มเข้าไปใหม่ ไม่เกิดการซ้ำซ้อนกับค่าเดิมของแอททริบิวท์เดียวกัน การเพิ่มค่ากระทำได้โดยป้อนคีย์ของแอททริบิวท์ที่ต้องการเพิ่มค่า และเพิ่มค่าโดยการรับข้อมูล ในกรณีที่ไม่มีแอททริบิวท์นั้นอยู่ หรือ กรณีที่ค่าที่เพิ่มเข้าไปใหม่เกิดการซ้ำซ้อนกับค่าเดิม จะมีการแสดงข้อผิดพลาดเกิดขึ้น

3.5.1.4.4 การลบค่าของแอททริบิวท์ (delete attribute value)

การลบค่าของแอททริบิวท์ จะทำการลบข้อเท็จจริง เมื่อไม่มีแอททริบิวท์อื่นใดใช้ค่านั้นอยู่ การลบกระทำได้โดยป้อนคีย์ของแอททริบิวท์ที่ต้องการลบค่า และเลือกรายการค่าที่ต้องการลบ ในกรณีที่ไม่มีแอททริบิวท์นั้นอยู่ จะแสดงข้อผิดพลาดเกิดขึ้น

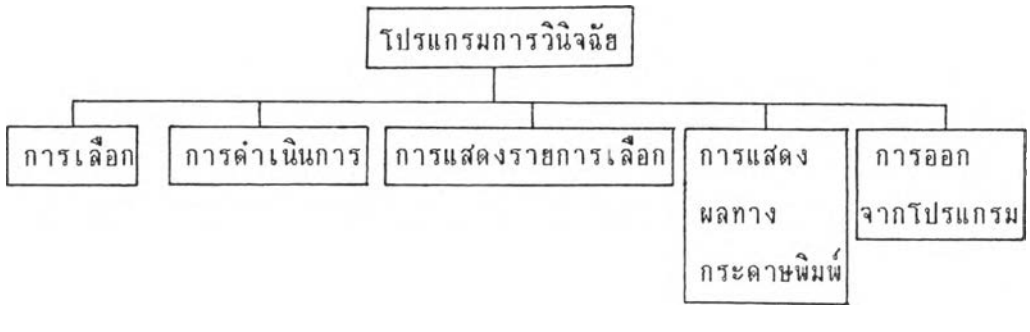
3.5.1.4.5 การเปลี่ยนแปลงการกระทำของกฎ (change rule action)

การกระทำของกฎจะแสดงเมื่อกฎนั้นถูกปฏิบัติการแล้ว และต้องการให้กระทำต่อไปอย่างไร ซึ่งส่วนกระทำของกฎจะใช้ในขั้นตอนการสรุปเหตุผล การเปลี่ยนแปลงการกระทำของกฎกระทำได้โดยป้อนเลขที่ของกฎที่ต้องการ ในกรณีที่ไม่มีหมายเลขกฎนั้นอยู่ จะมีการแสดงข้อผิดพลาดเกิดขึ้น

3.5.1.5 การจบการทำงานของโปรแกรม (quit)

เป็นการจบการทำงานของโปรแกรมเพื่อกลับเข้าสู่โปรแกรมจัดระบบงาน (operating system)

3.5.2 โปรแกรมการวินิจฉัย



รูปที่ 3.6 แสดงโปรแกรมการวินิจฉัย

ประกอบด้วย

3.5.2.1 การเลือก (select)

เริ่มต้นจากการเลือกระบบ หลังจากนั้นจะเลือกปัญหา (อาการ) จากระบบนั้น โดยจะมีรูปภาพส่วนของเครื่องยนต์ที่เกี่ยวข้องกับอาการในระบบ และสามารถเลื่อนไปภาพต่อไปได้ การเคาะรับภาพ จะแสดงสาเหตุที่อาจเกิดขึ้นกับอาการนั้นให้เลือกตอบตามอาการโดยมีการแสดงคำอธิบายและค่าให้เลือกพร้อมกับการตอบรับหรือตอบปฏิเสธ (positive or negative)

3.5.2.2 การดำเนินการ (run)

จะนำรายการเลือกที่เกิดจากขั้นตอนการเลือกไปประมวลผลตามแผนภาพต้นไม้ที่ได้สร้างขึ้นก่อนหน้านี้แล้ว และกระทำการค้นหาตามหลักการการค้นหาแบบหยั่งรู้ (heuristic search) จนกระทั่งสามารถสรุปคำตอบออกมาได้ พร้อมกับบอกการกระทำ (action) ของกฎที่ถูกดำเนินการนั้นเพื่อแนะนำข้อแก้ไข

3.5.2.3 การแสดงรายการเลือก (list)

เป็นการแสดงรายการที่เกิดจากการเลือกตอบจากรายการขั้นตอนการเลือก

3.5.2.4 การแสดงผลลัพธ์ทางกระดาษพิมพ์ (print)

เป็นการแสดงการสรุปคำตอบ และการแก้ไข จากขั้นตอนการดำเนินการ ออกทางเครื่องพิมพ์ (printer)

3.5.2.5 การออกจากโปรแกรม (quit)

เป็นการออกจากโปรแกรมการวินิจฉัย