

บทที่ 7

การสรุปความในตรรกศาสตร์อันดับหนึ่ง

(Inference in First-Order Logic)

เมื่อมีการแทนสิ่งต่าง ๆ ได้ด้วยตรรกศาสตร์อันดับหนึ่งแล้ว การสร้างกฎและการสรุปความทางตรรกะ ทำให้ได้ข้อสรุปใหม่เพิ่มเติมมากขึ้น ในบทนี้จะนำอัลกอริทึมมาช่วยให้สามารถตอบคำถามต่าง ๆ ถ้าคำถามเหล่านั้นหาคำตอบได้ เช่น มีความรู้เพียงพอ และอยู่ในขอบเขตของปัญหา

7.1 การสรุปความในตรรกศาสตร์ประพจน์กับตรรกศาสตร์อันดับหนึ่ง

เมื่อนำกฎการสรุปความแบบง่ายมาประยุกต์ใช้กับประโยคตรรกะที่มีตัวบ่งปริมาณ สามารถเปลี่ยนให้ประโยคเหล่านี้กลายเป็นประโยคที่ไม่มีตัวบ่งปริมาณ เมื่อเป็นเช่นนี้จึงทำให้เกิดแนวคิดที่ว่า เราสามารถทำการสรุปความกับตรรกศาสตร์อันดับหนึ่งได้ โดยการเปลี่ยนฐานความรู้ให้เป็นรูปแบบของตรรกศาสตร์ประพจน์ แล้วใช้วิธีการสรุปความตามแบบประพจน์ (ตามวิธีในบทที่ 5) จึงพิจารณาหาวิธีเหล่านี้มาปรับปรุงใช้สำหรับการจัดการกับตรรกศาสตร์อันดับหนึ่งโดยตรง

7.1.1 การใช้กฎการสรุปความกับตัวบ่งปริมาณ

กฎที่ใช้กับตัวบ่งปริมาณทั้งแบบ \forall และ \exists ได้แก่

7.1.1.1 Universal Instantiation

เมื่อมีประโยค α ใด ๆ ซึ่งประกอบด้วยตัวแปร v ให้แทนที่ตัวแปร v ทุกตัวในประโยคนั้นด้วยเทอมที่ไม่มีตัวแปร (Ground term) g เขียนเป็นรูปแบบของกฎได้ดังนี้

$$\frac{\forall v \quad \alpha}{\text{SUBST}(\{v | g\}, \alpha)}$$

โดยที่ $\text{SUBST}(\theta, \alpha)$ เป็นฟังก์ชันการแทนค่า (Substitution) หมายถึงการแทนค่า θ ในประโยค α ผลที่ได้คือประโยค α ใหม่หลังจากมีการแทนค่าแล้ว

การแทนค่า θ คือเซตของการแทนค่าสิ่งหนึ่งสิ่งใดด้วยค่าที่ต้องการ เช่น ถ้าเขียนว่า $\{v | g\}$ หมายถึงการแทนค่าตัวแปร v ด้วยค่า g

สมมติว่าฐานความรู้มีสัจพจน์กล่าวว่า “All greedy kings are evil”
เขียนเป็นตรรกะว่า $\forall x \text{ King}(x) \wedge \text{Greedy}(x) \Rightarrow \text{Evil}(x)$ จากประโยคนี้เมื่อมีการ
แจกแจงสมาชิกในโดเมนแล้ว สามารถสรุปได้ว่าประโยคต่าง ๆ เหล่านี้เป็นจริง

$$\text{King}(\text{John}) \wedge \text{Greedy}(\text{John}) \Rightarrow \text{Evil}(\text{John})$$

$$\text{King}(\text{Richard}) \wedge \text{Greedy}(\text{Richard}) \Rightarrow \text{Evil}(\text{Richard})$$

$$\text{King}(\text{Father}(\text{John})) \wedge \text{Greedy}(\text{Father}(\text{John})) \Rightarrow \text{Evil}(\text{Father}(\text{John}))$$

:

จากตัวอย่าง มีตัวแปร x อยู่ในประโยคตรรกะ การแทนค่า x ได้แก่การ
แทนค่า $\theta = \{ x \mid \text{John} \}, \{ x \mid \text{Richard} \}, \{ x \mid \text{Father}(\text{John}) \}$

7.1.1.2 Existential Instantiation

สำหรับประโยค α ใด ๆ ที่ประกอบด้วยตัวแปร v และสัญลักษณ์ค่าคงที่
(Constant symbol) k ที่ยังไม่เคยมีอยู่ในฐานความรู้มาก่อน ให้แทนที่ตัวแปร v ด้วย k เขียนเป็น
รูปแบบดังนี้

$$\frac{\exists v \quad \alpha}{\text{SUBST}(\{ v \mid k \}, \alpha)}$$

ตัวอย่างเช่น ประโยค

$$\exists x \text{ Crown}(x) \wedge \text{OnHead}(x, \text{John})$$

เมื่อใช้กฎ Existential Instantiation จะได้ข้อสรุปเป็นประโยคว่า

$$\text{Crown}(C_1) \wedge \text{OnHead}(C_1, \text{John})$$

เมื่อ C_1 เป็นสัญลักษณ์ค่าคงที่ ที่ไม่เคยมีอยู่ในฐานความรู้มาก่อน

ความหมายของกฎข้อนี้คือ ต้องการบอกให้ทราบว่า มีบางสิ่งบางอย่าง
(บาง Object) ที่เป็นไปตามเงื่อนไขที่ตั้งไว้ กระบวนการของกฎนี้สามารถให้ชื่อสิ่งนั้นออกมาได้
ชื่อนั้นต้องไม่เคยเป็นชื่อของสิ่งใดมาก่อน

ตัวอย่างทางคณิตศาสตร์ที่ทำให้เห็นได้ชัดเจน ได้แก่การที่พบว่า มีตัวเลข
ที่มีค่ามากกว่า 2.71828 อยู่เล็กน้อย และเป็นเลขที่สอดคล้องกับสมการ $\frac{d(x^y)}{dy} = x^y$ แล้ว

เราสามารถตั้งชื่อให้เลขจำนวนนี้ โดยให้ชื่อว่า e แต่เราไม่สามารถตั้งชื่อให้เข้ากับสิ่งที่มีอยู่แล้ว เช่น π ตรรกศาสตร์เรียกชื่อที่ตั้งใหม่โดยวิธีนี้ว่า Skolem constant

Existential Initiation เป็นกฎการสรุปความที่มีความซับซ้อนมากกว่า Universal Initiation วิธีการใช้กฎทั้ง 2 ข้อมีความแตกต่างกันเล็กน้อย กฎ Universal Initiation ใช้ซ้ำกันได้หลายครั้ง ทำให้เกิดประโยคต่าง ๆ กันเป็นจำนวนมาก ส่วน Existential Initiation ใช้ได้เพียงครั้งเดียวกับประโยคหนึ่ง หลังจากนั้น ประโยคที่มีตัวบ่งปริมาณ \exists นั้นจะไม่มี การนำมาใช้อีก เช่นประโยค $\exists x \text{ Kill}(x, \text{Victim})$

หลังจากมีการแทนที่เป็น $\text{Kill}(\text{Murderer}, \text{Victim})$ แล้ว ประโยคเดิมก็ไม่จำเป็นต้องใช้อีกต่อไป

7.1.1.3 การลดรูปไปเป็นการสรุปความในแบบประพจน์

หลังจากปรับรูปประโยคตรรกะแบบเดิม โดยตัดตัวบ่งปริมาณออกไปได้แล้ว กระบวนการสรุปความ สามารถดัดแปลงจากเดิมที่สรุปความตามแบบอันดับหนึ่ง (First-order inference) ไปเป็นการสรุปความตามแบบประพจน์ (Propositional inference) ได้แล้ว แนวคิดของการลดรูป (เรียกว่าลดรูปเพราะตรรกศาสตร์อันดับหนึ่งมีการแทนที่อย่างซับซ้อนมากกว่าตรรกศาสตร์ประพจน์) ทำได้โดยใช้ Existential Initiation หนึ่งครั้งกับประโยคที่มีตัวบ่งปริมาณ \exists และใช้ Universal Initiation หลายครั้งกับประโยคที่มีตัวบ่งปริมาณ \forall (ใช้กฎหลายครั้งเมื่อมีตัวแปรหลายตัว) เช่นสมมุติว่าฐานความรู้ประกอบด้วยประโยคต่อไปนี้

$$\forall x \text{ King}(x) \wedge \text{Greedy}(x) \Rightarrow \text{Evil}(x)$$

$\text{King}(\text{John})$

$\text{Greedy}(\text{John})$

$\text{Brother}(\text{Richard}, \text{John})$

ใช้กฎ Universal Initiation กับประโยคแรก โดยแทนตัวแปรด้วยสัญลักษณ์ค่าคงที่ทุกตัวที่มีในฐานความรู้ (ค่าคงที่เหล่านี้คือ Ground term) นั่นคือ $\{x \mid \text{John}\}$ และ $\{x \mid \text{Richard}\}$ จะได้ประโยคใหม่ที่ตัด \forall ออกไป 2 ประโยคดังนี้

$$\text{King}(\text{John}) \wedge \text{Greedy}(\text{John}) \Rightarrow \text{Evil}(\text{John})$$

$$\text{King}(\text{Richard}) \wedge \text{Greedy}(\text{Richard}) \Rightarrow \text{Evil}(\text{Richard})$$

หลังจากนี้ไม่ต้องใช้ประโยคเดิมที่มี \forall อีกต่อไป และถ้าพิจารณาประโยคเดี่ยว (Atomic sentence) ที่ประกอบขึ้นมาเป็นรูปประโยค ก็จะได้ประโยคเดี่ยวที่เป็นกราวนด์เทอม ได้แก่ $\text{King}(\text{John})$, $\text{Greedy}(\text{John})$ และอื่น ๆ ซึ่งเป็นสัญลักษณ์ประพจน์

(Proposition symbol) และสามารถใส่กระบวนการสรุปความตามแบบของตรรกศาสตร์ประพจน์ได้ (ดูบทที่ 5) และจะได้ข้อสรุปเป็น Evil(John)

7.2 ยูนิฟิเคชัน (Unification)

ยูนิฟิเคชันคือกระบวนการหาการแทนที่ที่ทำให้นิพจน์ตรรกะ 2 นิพจน์เหมือนกัน กระบวนการนี้ใช้มากในอัลกอริทึมของการสรุปความแบบอันดับหนึ่ง (First-order inference) ทุกรูปแบบ อัลกอริทึมของยูนิฟิเคชันมีชื่อว่า UNIFY มีหลักการทำงานคือ นำประโยค 2 ประโยค มาพิจารณา และหาการแทน (Unifier) ที่ทำให้ 2 ประโยคนั้นตรงกัน (ถ้ามี) มีรูปแบบการเขียนดังนี้

$$\text{UNIFY}(p, q) = \theta \text{ เมื่อ } \text{SUBST}(\theta, p) = \text{SUBST}(\theta, q)$$

ตัวอย่างเช่น ถ้ามีประโยคคำถาม

$$\text{Knows}(\text{Harry}, x)$$

เป็นประโยคที่ถามว่า แฮร์รี่รู้จักใครบ้าง การหาคำตอบคือการหาค่าตัวแปร x ทำได้โดยการหาทุกประโยคในฐานความรู้ที่ตรงกับประโยค $\text{Knows}(\text{Harry}, x)$ ผลที่ได้จากการทำยูนิฟิเคชันอาจเป็นประโยค 4 ประโยคที่อยู่ในฐานความรู้ ได้แก่

$$\text{UNIFY}(\text{Knows}(\text{Harry}, x), \text{Knows}(\text{Harry}, \text{Ron})) = \{x \mid \text{Ron}\}$$

$$\text{UNIFY}(\text{Knows}(\text{Harry}, x), \text{Knows}(y, \text{Sirius})) = \{x \mid \text{Sirius}, y \mid \text{Harry}\}$$

$$\text{UNIFY}(\text{Knows}(\text{Harry}, x), \text{Knows}(y, \text{Mother}(y))) = \{y \mid \text{Harry}, x \mid \text{Mother}(\text{Harry})\}$$

$$\text{UNIFY}(\text{Knows}(\text{Harry}, x), \text{Knows}(x, \text{Hermioni})) = \text{Fail}$$

ยูนิฟิเคชันในประโยคสุดท้ายได้ผลลัพธ์เป็น Fail หมายถึงหาผลลัพธ์ที่ต้องการไม่ได้ หรือทำงานไม่สำเร็จ เนื่องจาก x ไม่สามารถมีค่า 2 ค่าต่างกัน (คือ Harry กับ Hermioni) ในเวลาเดียวกันได้ เป็นที่สังเกตว่าประโยค $\text{Knows}(x, \text{Hermioni})$ มีความหมายว่า “ทุกคนรู้จักแฮร์รี่ไมโอની” จึงน่าจะสรุปได้ว่า แฮร์รี่ควรจะรู้จักแฮร์รี่ไมโอનીด้วย แต่ปัญหาที่เกิดขึ้นในที่นี้เป็นเพราะการใช้ตัวแปรเป็นชื่อเดียวกัน คือ x การเลี่ยงปัญหาเช่นนี้ทำได้โดยเปลี่ยนชื่อตัวแปรตัวใดตัวหนึ่ง เช่น

$$\text{UNIFY}(\text{Knows}(\text{Harry}, x), \text{Knows}(z, \text{Hermioni})) = \{x \mid \text{Hermioni}, z \mid \text{Harry}\}$$

ความซับซ้อนของยูนิฟิเคชันเกิดขึ้นได้เนื่องจากกระบวนการนี้ค้นหาการแทนค่าที่ทำให้ 2 ประโยคตรงกัน แต่บางครั้งอาจจะหาการแทนได้มากกว่า 1 แบบก็ได้ เช่น

$$\text{UNIFY}(\text{Knows}(\text{Harry}, x), \text{Knows}(y, z))$$

คำตอบที่ได้ อาจจะเป็น

$\{ y \mid \text{Harry}, x \mid z \}$ ประโยคที่เกิดจากยูนิฟิเคชันคือ $\text{Knows}(\text{Harry}, z)$

$\{ y \mid \text{Harry}, x \mid \text{Harry}, z \mid \text{Harry} \}$ ประโยคที่เกิดจากยูนิฟิเคชันคือ

$\text{Knows}(\text{Harry}, \text{Harry})$

ผลการแทนประโยคที่ 2 เกิดขึ้นได้เมื่อมีการแทนค่าเพิ่มเติม $\{ z \mid \text{Harry} \}$ ลงในการแทนประโยคแรก การแทนประโยคนั้นมีลักษณะทั่วไปมากกว่าประโยคที่ 2 เพราะมีการแทนค่าตัวแปรที่ไม่เฉพาะเจาะจง และมีข้อจำกัดน้อยกว่าในประโยคที่ 2 สรุปว่าในการทำยูนิฟิเคชันประโยคแต่ละคู่ จะมีการแทนอยู่ในรูปทั่วไปมากที่สุดได้อันหนึ่ง เรียกว่า Most general unifier (MGU) การแทนอื่นสามารถเกิดจากการตั้งชื่อใหม่ให้กับตัวแปรใน MGU นี้ เช่นตัวอย่างที่อธิบายมาแล้ว มี MGU คือ $\{ y \mid \text{Harry}, x \mid z \}$

ผลจากการทำยูนิฟิเคชันที่สำคัญข้อหนึ่งคือ ทำให้ได้การแทนตัวแปรต่าง ๆ เช่น เมื่อแทนค่าตัวแปรในประโยค Implication ในส่วนของ Premise แล้ว จะทำให้ได้ผลสรุปในส่วน Conclusion ซึ่งเป็นวิธีการใช้กฎการสรุปความ Modus Ponens ที่พบเห็นมากในกระบวนการสรุปความส่วนใหญ่

กฎการสรุปความที่อยู่ในรูปทั่วไปมากกว่า Modus Ponens คือ Generalized Modus Ponens (GMP) กล่าวว่า สำหรับประโยคเดียว $p_i, p_i',$ และ q มีการแทนค่า θ ซึ่ง $\text{SUBST}(\theta, p_i') = \text{SUBST}(\theta, p_i)$ สำหรับทุก i หรือเขียนว่า

$$\frac{p_1', p_2', \dots, p_n', (p_1 \wedge p_2 \wedge \dots \wedge p_n \Rightarrow q)}{\text{SUBST}(\theta, q)}$$

ตัวอย่าง

p_1' คือ $\text{King}(\text{John})$

p_1 คือ $\text{King}(x)$

p_2' คือ $\text{Greedy}(y)$

p_2 คือ $\text{Greedy}(x)$

θ คือ $\{ x \mid \text{John}, y \mid \text{John} \}$

q คือ $\text{Evil}(x)$

$\text{SUBST}(\theta, q)$ คือการแทน θ ในประโยค q ผลที่ได้คือ $\text{Evil}(\text{John})$

Generalized Modus Ponens มีความเป็นทั่วไปมากกว่า Modus Ponens ธรรมดา และลดขั้นตอนการสรุปความให้สั้นลงได้ เพราะสามารถแทนตัวแปรหลายตัวในประโยคต่าง ๆ ได้ในขั้นตอนเดียวเท่านั้น

7.3 อัลกอริทึมแบบฟอร์เวิร์ดเชนนิ่ง (Forward Chaining Algorithm)

เป็นการนำอัลกอริทึมมาใช้กับประโยคตรรกะ เพื่อหาข้อสรุปใหม่จากประโยคเดิม อัลกอริทึมแบบฟอร์เวิร์ดเชนนิ่งมีลักษณะการทำงานตรงไปตรงมา และง่าย โดยเริ่มจากประโยคเดี่ยวต่าง ๆ ที่อยู่ในฐานความรู้นำมาใช้ร่วมกับกฎการสรุปความ Modus Ponens ทำให้เกิดข้อสรุปใหม่ที่เป็นประโยคเดี่ยวเพิ่มขึ้น ประโยคที่จะใช้อัลกอริทึมแบบนี้เป็นประโยค Implication ที่สามารถให้ข้อสรุปใหม่ได้ มักจะอยู่ในรูปของ

Situation \Rightarrow Response

นั่นคือ ทางซ้ายมือของประโยค หรือ Premise เป็นตัวบอกสถานการณ์หรือเงื่อนไขที่จะทำให้เกิดการตอบสนองตามกฎเกณฑ์ที่วางไว้ หรือก็คือข้อสรุป (Conclusion) ทางขวามือนั่นเอง

ประโยค Implication ที่ใช้ในอัลกอริทึมนี้ ต้องเขียนได้ในรูปของ Horn clause เสมอ นั่นคือ จะเป็น Disjunction ของค่าข้อความ (Literal) โดยมีเพียงค่าเดียวเท่านั้นที่เป็นเชิงบวก (เป็น Positive ค่าเดียว นอกนั้นเป็น Negative หหมด) ลักษณะนี้ทำให้เขียนอยู่ในรูป Implication ได้ โดยทางซ้ายมือเป็น Conjunction ของค่าข้อความที่เป็นบวก ส่วนผลสรุปทางขวามือเป็นค่าข้อความเดี่ยวที่เป็นบวก (บวกในความหมายของตรรกะหมายถึง ไม่เป็นนิเสธ) ประโยคที่ไม่ใช่รูป Implication ที่นำมาใช้ในอัลกอริทึมสามารถเป็นประโยคเดี่ยวได้ ตัวอย่างของประโยคที่ใช้ได้ ได้แก่

$\text{King}(x) \wedge \text{Greedy}(x) \Rightarrow \text{Evil}(x)$

$\text{King}(\text{John})$

$\text{Greedy}(y)$

ค่าข้อความที่มีตัวแปรติด ตัวแปรเหล่านั้นมักจะมี ความหมายเหมือนกับว่าใช้ตัวบ่งปริมาณ \forall แต่จะละไว้โดยไม่ต้องเขียนตัวบ่งปริมาณกำกับ

ฐานความรู้ส่วนใหญ่สามารถแปลงเป็นประโยคตรรกะที่เหมาะสมกับการใช้อัลกอริทึมนี้ได้ เช่นตัวอย่างต่อไปนี้

“The law says that it is a crime for an American to sell weapons to hostile nations. The country Nono, an enemy of America, has some missiles, and all of its missiles were sold to it by Colonel West, who is American.”

ต้องการพิสูจน์ว่า “West is a criminal” ก่อนอื่นจะต้องแทนข้อเท็จจริงต่าง ๆ ให้เป็นประโยคตรรกะดังนี้

1. “. . . it is a crime for an American to sell weapons to hostile nations”

$$\text{American}(x) \wedge \text{Weapon}(y) \wedge \text{Nation}(z) \wedge \text{Hostile}(z)$$

$$\wedge \text{Sells}(x, y, z) \Rightarrow \text{Criminal}(x) \quad (1)$$

2. “Nono... has some missiles” ประโยคนี้ถ้าเขียนพร้อมตัวบ่งปริมาณจะได้ว่า

$$\exists x \text{ Owns}(\text{Nono}, x) \wedge \text{Missile}(x)$$

ให้ใช้ Existential Initiation ตัด \exists ทิ้ง แล้วแทนค่า x ด้วยกราวนด์เทอม ดังนี้

$$\text{Owns}(\text{Nono}, M1) \quad (2)$$

$$\text{Missile}(M1) \quad (3)$$

3. “All of its missiles were sold to it by Colonel West”

$$\text{Missile}(x) \wedge \text{Owns}(\text{Nono}, x) \Rightarrow \text{Sells}(\text{West}, x, \text{Nono}) \quad (4)$$

4. จากข้อเท็จจริงเกี่ยวกับ Missile จำเป็นต้องมีกฎบอกให้รู้ว่าจรวดเป็นอาวุธ ดังนี้

$$\text{Missile}(x) \Rightarrow \text{Weapon}(x) \quad (5)$$

5. ข้อเท็จจริงว่าศัตรูของอเมริกาเรียกว่า “Hostile”

$$\text{Enemy}(x, \text{America}) \Rightarrow \text{Hostile}(x) \quad (6)$$

6. “West, who is American...”

$$\text{American}(\text{West}) \quad (7)$$

7. “The country Nono, an enemy of America ...”

$$\text{Enemy}(\text{Nono}, \text{America}) \quad (8)$$

$$\text{Nation}(\text{Nono}) \quad (9)$$

$$\text{Nation}(\text{America}) \quad (10)$$

อัลกอริทึมแบบฟอร์เวิร์ดเชนนิ่งนำข้อเท็จจริงในฐานความรู้มาใช้หาว่า Premise ของประโยค Implication เป็นจริง แล้วจะได้ส่วนผลสรุป (Conclusion) เป็นจริง ส่วนผลสรุปนี้ นำมาใช้เป็นข้อเท็จจริงเพิ่มเติม เพื่อค้นหาผลสรุปของประโยคอื่น ๆ อีก จนกว่าจะไม่มีผลสรุปใด เกิดใหม่อีกแล้ว หรือจนกว่าจะได้รับคำตอบที่ต้องการหา

ขั้นตอนการทำงานของฟอร์เวิร์ดเชนนิ่งเพื่อการพิสูจน์ว่า $\text{Criminal}(\text{West})$ เป็นจริง ต้องพิจารณาประโยค Implication ในข้อ (1), (4), (5), (6) ตามลำดับ ดังนี้

1. จาก (1) Premise ในข้อนี้ยังหาค่าความจริงไม่ได้ เนื่องจากข้อเท็จจริงเท่าที่มีอยู่ไม่เพียงพอที่จะนำมาใช้ Premise ในข้อนี้ประกอบด้วย

$$\text{American}(x) \wedge \text{Weapon}(y) \wedge \text{Sells}(x, y, z) \wedge \text{Hostile}(z)$$

ขณะนี้ แม้ว่าจะแทนค่าตัวแปร x ด้วย West แต่ก็ทำให้ได้ประโยคเดียว American(West) เป็นจริงเพียงประโยคเดียว แต่ยังไม่รู้ว่า $\text{Weapon}(y) \wedge \text{Sells}(\text{West}, y, z) \wedge \text{Hostile}(z)$ เป็นจริงหรือไม่

2. จาก (4) เมื่อแทนค่า x ด้วย M1 แล้ว Premise ในข้อนี้คือ

$$\text{Missile}(M1) \wedge \text{Owns}(\text{Nono}, M1)$$

โดยใช้กฎการสรุปความ GMP จะทำให้ได้ผลสรุปเป็นจริง สามารถเติมประโยค

$$\text{Sells}(\text{West}, \text{Nono}, M1)$$

ลงไปในฐานความรู้ เป็นข้อเท็จจริงเพิ่มเติมได้

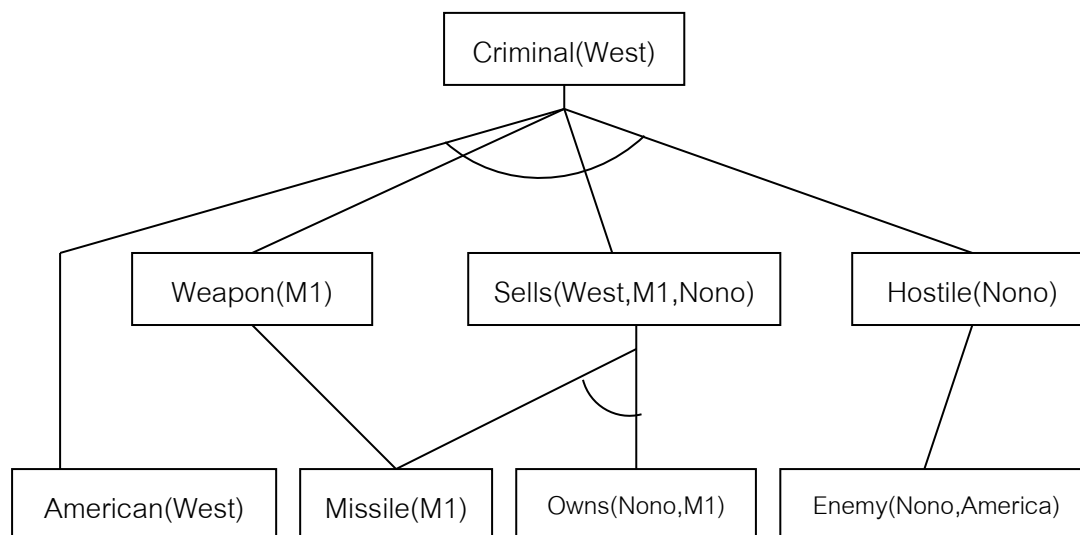
3. จาก (5) เมื่อแทนค่า x ด้วย M1 ทำให้ Premise ในข้อนี้เป็นจริง จึงได้ผลสรุปว่า $\text{Weapon}(M1)$ เป็นจริง และเติมลงฐานความรู้เป็นข้อเท็จจริงเพิ่มเติม

4. จาก (6) เมื่อแทนค่า x ด้วย Nono ทำให้ Premise ในข้อนี้เป็นจริง จึงได้ผลสรุปว่า $\text{Hostile}(\text{Nono})$ เป็นจริง และเติมลงฐานความรู้เป็นข้อเท็จจริงเพิ่มเติม

5. จากข้อเท็จจริงใหม่ที่ได้มาในขั้นตอนที่ 2-4 ข้างต้น ใช้กฎการสรุปความ GMP จะได้ผลสรุปว่า

$\text{Criminal}(\text{West})$ เป็นจริง

ฟอร์เวิร์ดเชนนิ่งสามารถเขียนแผนผังเป็น Proof tree ได้ดังรูปที่ 7.1



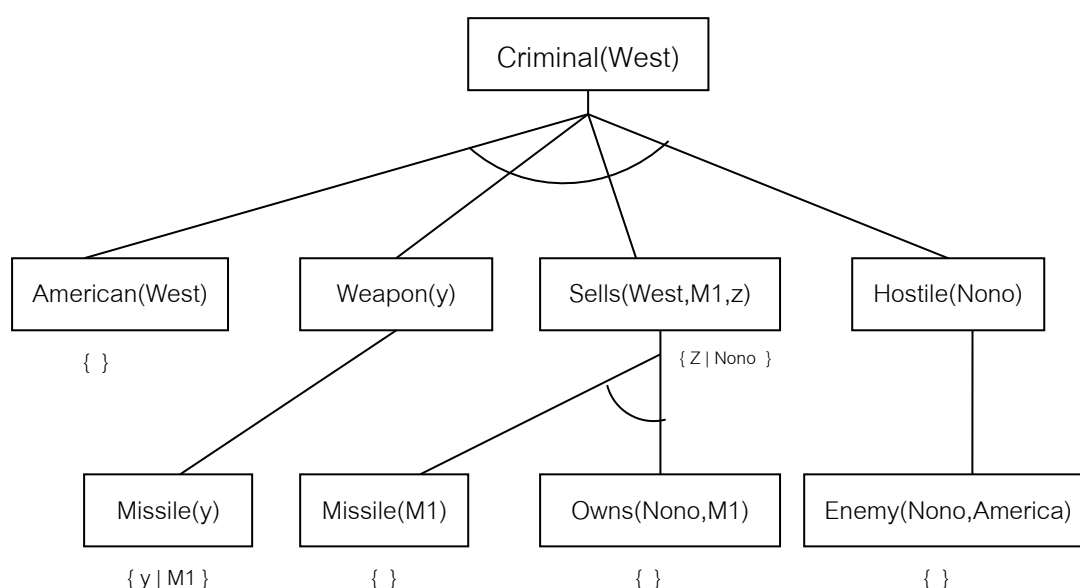
รูปที่ 7.1 แผนผัง Proof tree ของการพิสูจน์โดยใช้อัลกอริทึมแบบฟอร์เวิร์ดเชนนิ่ง

จากรูปที่ 7.1 ชั้นล่างสุดเป็นข้อเท็จจริงที่มีอยู่เดิมในฐานความรู้ ชั้นกลางเป็นข้อเท็จจริงใหม่ที่เกิดจากการสรุปความจากข้อเท็จจริงชั้นล่าง ส่วนชั้นบนสุดเป็นข้อเท็จจริงที่เกิดจากการสรุปความจากข้อเท็จจริงชั้นกลาง

7.4 อัลกอริทึมแบบแบ็คเวิร์ดเชนนิ่ง (Backward chaining algorithm)

เป็นอัลกอริทึมที่ทำงานเริ่มจากเป้าหมายแล้วย้อนกลับมา สืบสาวกลับจนพบข้อเท็จจริงที่สนับสนุนการพิสูจน์ได้ครบสมบูรณ์ การทำงานตลอดทั้งกระบวนการเป็นแบบแบ็คเวิร์ดเชนนิ่งทั้งสิ้น บางครั้งเรียกรวีนี้อีกว่า Goal-directed reasoning

การทำงานของอัลกอริทึมนี้จะค้นหาคำถาม (Query) ที่เป็นเป้าหมายที่ต้องการหาตอบ หรือต้องการพิสูจน์ โดยหาจากส่วนผลสรุปที่อยู่ทางขวามือของประโยค Implication เมื่อพบผลสรุปที่ตรงกันกับคำถามแล้ว จะย้อนกลับมาที่ส่วน Premise ที่อยู่ทางซ้ายมือ ถ้าทุกประโยคใน Premise เป็นความจริง การพิสูจน์จะสำเร็จ และสามารถเพิ่มคำตอบที่หาได้ลงในฐานความรู้ ตัวอย่างการทำงานโดยใช้อัลกอริทึมเช่นนี้กับตัวอย่าง Criminal(West) แสดงได้ในรูปที่ 7.2



รูปที่ 7.2 แผนผัง Proof tree ของการพิสูจน์โดยอัลกอริทึมแบบแบ็คเวิร์ดเชนนิ่ง

การพิสูจน์โดยวิธีแบ็คเวิร์ดเชนนิ่งตามรูปที่ 7.2 เริ่มต้นจากเป้าหมายที่ต้องการพิสูจน์คือ Criminal(West) เมื่อค้นหาในฐานความรู้ พบว่ามีประโยค Implication ที่มีคำว่า Criminal(x) เป็นส่วนผลสรุปในประโยค (1) จึงนำแต่ละประโยคเดียวใน Premise มาเรียงกันในระดับชั้นล่าง

ต่อมา (ตามรูปเป็นชั้นกลาง) เริ่มพิจารณาประโยคในชั้นนี้จากซ้ายไปขวาตามลำดับ เพื่อพิสูจน์ว่าแต่ละประโยคเป็นความจริงหรือไม่ โดยวิธีการแบ็คเวิร์ดเช่นเดียวกัน ในระดับชั้นนี้มีขั้นตอนการพิจารณาดังนี้

1. ประโยค American(West) มีค่าความจริงเป็นจริงจาก (7) ในฐานความรู้ การเขียนเครื่องหมายวงเล็บ { } กำกับไว้ข้างล่างรูปสี่เหลี่ยม แสดงว่าไม่มีการแทนค่าตัวแปรใด ๆ

2. ประโยค Weapon(y) ไม่ใช่ข้อเท็จจริงที่เป็นประโยคเดียวอยู่ในฐานความรู้ จึงใช้กระบวนการแบ็คเวิร์ดค้นหา และพบว่า ประโยคนี้นปรากฏเป็นผลสรุปอยู่ใน (5) จึงย้อนกลับไปที่ส่วน Premise ของประโยค ได้แก่

Missile(y)

เมื่อแทนค่า y ด้วย M1 จะพบประโยค Missile(M1) เป็นจริงอยู่ในฐานความรู้ (3) เมื่อส่วน Premise เป็นจริง จะได้ Weapon(y) เป็นจริง

3. ประโยค Sells(West, M1, z) ไม่ใช่ข้อเท็จจริงที่เป็นประโยคเดียวอยู่ในฐานความรู้ จึงใช้กระบวนการแบ็คเวิร์ดค้นหา และพบว่า ประโยคนี้นปรากฏเป็นผลสรุปอยู่ใน (4) จึงย้อนกลับไปที่ส่วน Premise ของประโยค ได้แก่

Missile(M1) \wedge Owns(Nono, M1)

Missile(M1) เป็นจริงตามข้อเท็จจริง (3) ส่วน Owns(Nono, M1) เป็นจริงตามข้อเท็จจริง (2) เมื่อแทนค่า z ด้วย Nono ทำให้ได้ Sells(West, M1, Nono) เป็นจริง

4. ประโยค Hostile(Nono) ไม่ใช่ข้อเท็จจริงที่เป็นประโยคเดียวอยู่ในฐานความรู้ จึงใช้กระบวนการแบ็คเวิร์ดค้นหา และพบว่า ประโยคนี้นปรากฏเป็นผลสรุปอยู่ใน (6) จึงย้อนกลับไปที่ส่วน Premise ของประโยค ได้แก่

Enemy(Nono, America)

ประโยคนี้เป็นจริง (8) จึงสรุปได้ว่า Hostile(Nono) เป็นจริง

5. เมื่อทุกประโยคในระดับชั้นนี้เป็นจริง ย้อนกลับไปยังชั้นที่สูงกว่า คือ Criminal(West) จะเป็นจริงด้วย

7.5 ริโซลูชัน (Resolution)

กระบวนการทำงานของริโซลูชัน ใช้หลักการสรุปความให้เป็นนิเสธ แล้วพิสูจน์ให้เกิดการขัดแย้งขึ้น เพื่อให้ได้ผลสรุปที่ต้องการ เรียกว่าการพิสูจน์โดยข้อขัดแย้ง (Proof by contradiction)

7.5.1 การแปลงประโยคให้อยู่ในรูป CNF (Conjunctive normal form)

ประโยคที่จะใช้วิธีพิสูจน์ได้ต้องอยู่ในรูป CNF นั่นคือ เป็น Conjunction ของวลีต่าง ๆ (วลีที่เชื่อมกันด้วย \wedge) โดยแต่ละวลีจะเป็น disjunction ของค่าข้อความ ค่าข้อความอาจมีตัวแปรอยู่ด้วยก็ได้ ตัวแปรเหล่านี้ต้องมีความหมายของการบ่งปริมาณแบบ \forall แต่ละไว้โดยไม่ต้องเขียน ตัวอย่างเช่น ประโยค

$$\forall x, y, z \text{ American}(x) \wedge \text{Weapon}(y) \wedge \text{Sells}(x, y, z) \wedge \text{hostile}(z) \Rightarrow \text{Criminal}(x)$$

เขียนในรูป CNF ได้ว่า

$$\neg \text{American}(x) \vee \neg \text{Weapon}(y) \vee \neg \text{Sells}(x, y, z) \vee \neg \text{hostile}(z) \vee \text{Criminal}(x)$$

ประโยคทุกประโยคในตรรกศาสตร์อันดับหนึ่งสามารถแปลงให้เป็น CNF ได้ทั้งสิ้น และเนื่องจากการสรุปความโดยใช้ประโยคตรรกศาสตร์อันดับหนึ่งเป็นเช่นเดียวกันกับประโยคที่อยู่ในรูป CNF จึงสามารถใช้ CNF ในการพิสูจน์แบบ Contradiction ได้เช่นเดียวกัน

กระบวนการขั้นตอนของการแปลงประโยคให้อยู่ในรูป CNF ดูได้จากตัวอย่างประโยคต่อไปนี้

“Everyone who loves all animals is loved by someone.”

ประโยคนี้นี้เขียนเป็นตรรกะว่า

$$\forall x [\forall y \text{ Animal}(y) \Rightarrow \text{Loves}(x, y)] \Rightarrow [\exists z \text{ Loves}(z, x)]$$

ขั้นตอนการแปลงเป็น CNF ได้แก่

1. แปลงประโยคไม่ให้มี Implication

$$\forall x [\neg \forall y \neg \text{Animal}(y) \vee \text{Loves}(x, y)] \vee [\exists z \text{ Loves}(z, x)]$$

2. ย้าย \neg โดยใช้กฎสำหรับนิเสธกับการบ่งปริมาณดังนี้

$$\neg \forall x P \text{ เปลี่ยนเป็น } \exists x \neg P$$

$$\neg \exists x P \text{ เปลี่ยนเป็น } \forall x \neg P$$

ประโยคมีการเปลี่ยนแปลง โดยแสดงทีละขั้นตอนได้ 3 ประโยคต่อไปนี้

$$\forall x [\exists y \neg (\neg \text{Animal}(y) \vee \text{Loves}(x, y))] \vee [\exists z \text{ Loves}(z, x)]$$

$$\forall x [\exists y \neg \neg \text{Animal}(y) \wedge \neg \text{Loves}(x, y)] \vee [\exists z \text{ Loves}(z, x)]$$

$$\forall x [\exists y \text{ Animal}(y) \wedge \neg \text{Loves}(x, y)] \vee [\exists z \text{ Loves}(z, x)]$$

จะสังเกตเห็นว่า ตัวบ่งปริมาณ $\forall y$ ในส่วน Premise ของประโยค Implication เปลี่ยนเป็นตัวบ่งปริมาณ $\exists y$ อ่านประโยคได้ว่า “Either there is some animal that x doesn't love, or someone loves x” แต่ความหมายยังคงเดิม ไม่เปลี่ยนแปลง

3. ตัดตัวบ่งปริมาณ \exists ออก กระบวนการนี้คล้ายกับการใช้กฎการสรุปความ Existential Instantiation ซึ่งแปลงประโยค $\exists x P(x)$ ให้เป็น $P(A)$ เมื่อ A คือค่าคงที่ตัวใหม่ แต่ถ้าใช้กฎข้อนี้ ประโยคที่ได้คือ

$$\forall x [\text{Animal}(A) \wedge \neg \text{Loves}(x, A)] \vee \text{Loves}(B, x)$$

ซึ่งมีความหมายผิดไปจากเดิมโดยสิ้นเชิง เพราะประโยคใหม่นี้กล่าวว่า “Everyone either fails to love a particular animal A or is loved by some particular entity B” แต่ประโยคเดิมหมายความว่า กำหนดให้แต่ละคนนั้น ไม่รักสัตว์ใด ๆ (ที่ไม่ใช่สัตว์ตัวเดียวกัน) หรือไม่ก็ได้รับความรักจากบุคคลใด ๆ (ที่ไม่ใช่คนเดียวกัน) ดังนั้นจึงต้องหาฟังก์ชันที่ทำให้ได้ค่าที่ขึ้นอยู่กับ x มาแทน เรียกว่า Skolem function

$$\forall x [\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))] \vee \text{Loves}(G(x), x)$$

F และ G คือ Skolem function มีกฎการใช้ฟังก์ชันเหล่านี้ว่า Argument ใด ๆ ที่อยู่ ในฟังก์ชัน จะมีตัวบ่งปริมาณแบบ \forall

4. ตัดตัวบ่งปริมาณ \forall ออกไป เมื่อถึงขั้นตอนนี้ ตัวแปรทุกตัวมีตัวบ่งปริมาณแบบ \forall ทั้งสิ้น จะได้ประโยคใหม่ คือ

$$[\text{Animal}(F(x)) \wedge \neg \text{Loves}(x, F(x))] \vee \text{Loves}(G(x), x)$$

5. กระจาย \wedge ไปบน \vee

$$[\text{Animal}(F(x)) \vee \text{Loves}(G(x), x)] \wedge [\neg \text{Loves}(x, F(x)) \vee \text{Loves}(G(x), x)]$$

ประโยคในขั้นตอนที่ 5 เป็น Conjunction ของวลีที่เป็น Disjunction และอยู่ในรูปของ CNF อาจจะอ่านแล้วทำความเข้าใจความหมายยาก สำหรับมนุษย์ไม่จำเป็นต้องใช้รูปแบบประโยคเช่นนี้ แต่การแปลงเป็นรูปแบบนี้ทำให้อัลกอริทึมทางตรรกะทำงานได้ง่ายขึ้น

7.5.2 กฎการสรุปความแบบปริิโซลูชัน

กฎการสรุปความแบบปริิโซลูชันกับวลีที่เป็นตรรกศาสตร์อันดับหนึ่ง ต้องมี 2 วลี ทั้ง 2 วลีไม่ใช่ตัวแปรร่วมกัน ค่าข้อความที่เป็นตรรกศาสตร์อันดับหนึ่ง ต้องมีค่าหนึ่งที่มีการแทนตรงกันกับนิเสธของค่าข้อความอีกค่าหนึ่ง รูปแบบของกฎเป็นดังนี้

$$\frac{l_1 \vee \dots \vee l_k, m_1 \vee \dots \vee m_n}{\text{SUBST}(\theta, l_1 \vee \dots \vee l_{i-1} \vee l_{i+1} \vee \dots \vee l_k \vee m_1 \vee \dots \vee m_{j-1} \vee m_{j+1} \vee \dots \vee m_n)}$$

เมื่อ $\text{UNIFY}(l_i, \neg m_j) = \theta$

ตัวอย่างการทำรีโซลูชัน วลี 2 วลี ดังนี้

$$[\text{Animal}(F(x)) \vee \text{Loves}(G(x), x)] \text{ และ}$$

$$[\neg \text{Loves}(u, v) \vee \neg \text{Kills}(u, v)]$$

ตัดค่าข้อความที่เป็นนิเสธของกันและกัน คือ $\text{Loves}(G(x), x)$ และ $\neg \text{Loves}(u, v)$

โดยมีการแทน $\theta = \{ u | G(x), v | x \}$ ผลที่ได้จากการทำรีโซลูชันคือ

$$[\text{Animal}(F(x)) \vee \neg \text{Kills}(G(x), x)]$$

ตัวอย่างการพิสูจน์โดยวิธีรีโซลูชัน มี 2 ตัวอย่าง

ตัวอย่างที่ 1 $\text{Criminal}(\text{West})$ มีขั้นตอนการพิสูจน์ตามขั้นตอนแผนภาพในรูปที่ 7.3

ประโยคต่าง ๆ ในตัวอย่างเขียนเป็นประโยค CNF ได้ดังนี้

$$\neg \text{American}(x) \vee \neg \text{Weapon}(y) \vee \neg \text{Sells}(x, y, z) \vee \neg \text{Hostile}(z) \vee \text{Criminal}(x)$$

$$\neg \text{Missile}(x) \vee \neg \text{Owns}(\text{Nono}, x) \vee \text{Sells}(\text{West}, x, \text{Nono})$$

$$\neg \text{Enemy}(x, \text{America}) \vee \text{hostile}(x)$$

$$\neg \text{Missile}(x) \vee \text{Weapon}(x)$$

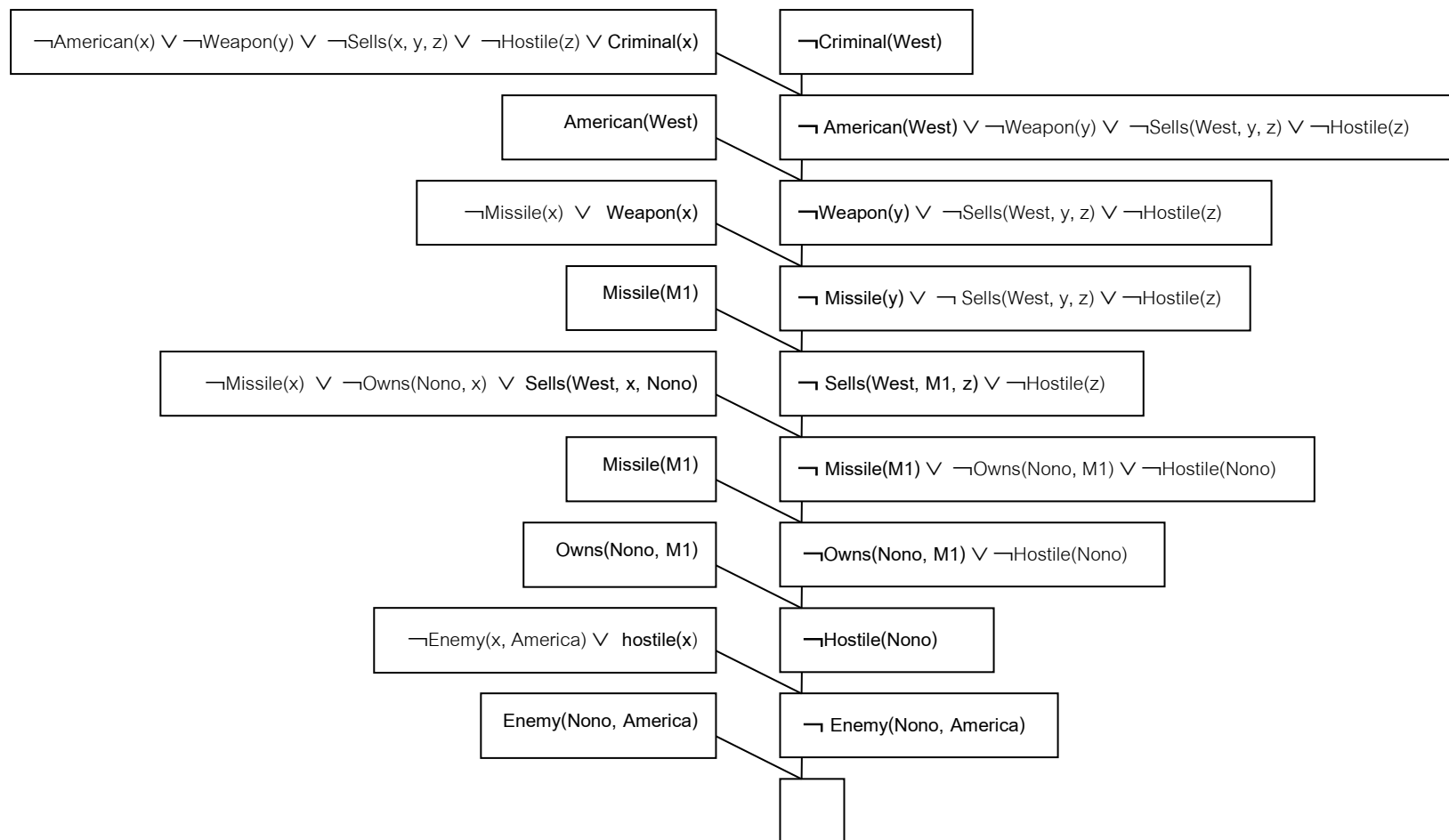
$$\text{Owns}(\text{Nono}, \text{M1})$$

$$\text{Missile}(\text{M1})$$

$$\text{American}(\text{West})$$

$$\text{Enemy}(\text{Nono}, \text{America})$$

การพิสูจน์แบบวิธีรีโซลูชัน กำหนดให้เป้าหมายเป็นนิเสธ คือ $\neg \text{Criminal}(\text{West})$ โดยที่นำประโยคนี้ออกไปเก็บไว้ในฐานความรู้ก่อน แผนภาพในรูปที่ 7.3 มีลักษณะเหมือนกระดุกสนหลังประโยคในกล่องแต่ละคู่ที่พิมพ์ตัวหนา มีการสลายข้อ (Resolve) กันโดยพิจารณาจากความรู้ที่นำมาจากฐานความรู้ ทำไปเรื่อยจนกว่าจะได้วลีว่างในกล่องล่างสุด ซึ่งแสดงว่าเกิดการขัดแย้งขึ้นกับเป้าหมายที่วางไว้ จึงต้องปฏิเสธเป้าหมาย นั่นคือกลับจาก $\neg \text{Criminal}(\text{West})$ เป็น $\text{Criminal}(\text{West})$



รูปที่ 7.3 การพิสูจน์แบบปริไซลูชันว่า West เป็น Criminal

ตัวอย่างที่ 2 กำหนดประโยคต่อไปนี้

Everyone who loves all animals is loved by someone.

Anyone who kills an animal is loved by no one.

Jack loves all animals.

Either Jack or Curiosity killed the cat, who is named Tuna.

Did Curiosity kill the cat?

ประโยคเหล่านี้แปลงเป็นตรรกศาสตร์อันดับหนึ่งพร้อมกับตัวบ่งปริมาณได้ดังนี้

A. $\forall x [\forall y \text{ Animal}(y) \Rightarrow \text{Loves}(x, y)] \Rightarrow [\exists z \text{ Loves}(z, x)]$

B. $\forall x [\exists y \text{ Animal}(y) \wedge \text{Kills}(x, y)] \Rightarrow [\forall z \neg \text{Loves}(z, x)]$

C. $\forall x \text{ Animal}(x) \Rightarrow \text{Loves}(\text{Jack}, x)$

D. $\text{Kills}(\text{Jack}, \text{Tuna}) \vee \text{Kills}(\text{Curiosity}, \text{Tuna})$

E. $\text{Cat}(\text{Tuna})$

F. $\forall x \text{ Cat}(x) \Rightarrow \text{Animal}(x)$

\neg G. $\neg \text{Kills}(\text{Curiosity}, \text{Tuna})$

ข้อสุดท้ายคือ \neg G เป็นการให้เป้าหมายที่เป็นนิเสธ เพื่อก่อให้เกิดความขัดแย้งขึ้นภายหลังตามหลักของการพิสูจน์ Proof by contradiction ต่อไปแปลงประโยคให้เป็นรูป CNF ได้ดังนี้

A1. $\text{Animal}(F(x)) \vee \text{Loves}(G(x), x)$

A2. $\neg \text{Loves}(x, F(x)) \vee \text{Loves}(G(x), x)$

B. $\neg \text{Animal}(y) \vee \neg \text{Kills}(x, y) \vee \neg \text{Loves}(z, x)$

C. $\neg \text{Animal}(x) \vee \text{Loves}(\text{Jack}, x)$

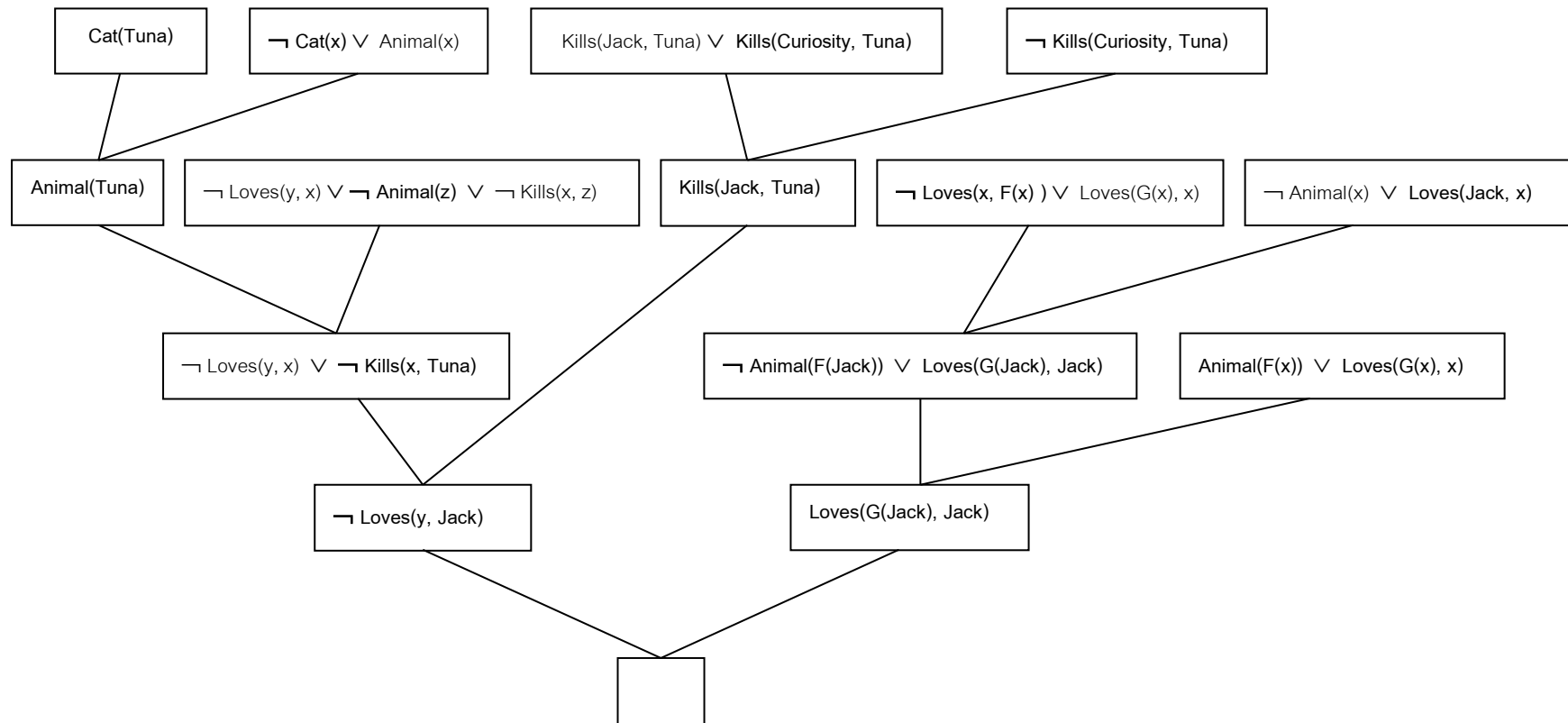
D. $\text{Kills}(\text{Jack}, \text{Tuna}) \vee \text{Kills}(\text{Curiosity}, \text{Tuna})$

E. $\text{Cat}(\text{Tuna})$

F. $\neg \text{Cat}(\text{Tuna}) \vee \text{Animal}(x)$

\neg G. $\neg \text{Kills}(\text{Curiosity}, \text{Tuna})$

การพิสูจน์แบบวิธีพิสูจน์ว่า Curiosity kills the cat แสดงเป็นแผนภาพไว้ในรูปที่ 7.4 การพิสูจน์นี้ตอบคำถามว่า “Curiosity ซ่าแมวหรือไม่” แต่ถ้าต้องการถามคำถามที่กว้างขวางขึ้น เช่นถามว่า “ใครซ่าแมว” วิธีการวิธีพิสูจน์ก็สามารถหาคำตอบให้ได้ แต่ต้องใช้เวลาในการทำงานเพิ่มขึ้นโดยพยายามเชื่อมโยงค่าให้กับตัวแปรขณะที่พิสูจน์



รูปที่ 7.4 การพิสูจน์แบบปริทัศน์ว่า "Curiosity kills the cat"

แบบฝึกหัดบทที่ 7

1. จากประโยคเดี่ยวแต่ละคู่ต่อไปนี้ จงหาการแทนที่ทั่วไปที่สุดที่มี (Most general unifier)
 - 1.1 $P(A, B, B), P(x, y, z)$
 - 1.2 $Q(y, G(A, B)), Q(G(x, x), y)$
 - 1.3 $\text{Older}(\text{Father}(y), y), \text{Older}(\text{Father}(x), \text{John})$
 - 1.4 $\text{Knows}(\text{Father}(y), y), \text{Knows}(x, x)$
2. จงเขียนประโยคต่อไปนี้ให้อยู่ในรูปตรรกศาสตร์อันดับหนึ่งที่เหมาะสมกับการใช้กับกฎ Generalized Modus Ponens ต่อไป
 - 2.1 Horses, cows, and pigs are mammals.
 - 2.2 An offspring of a horse is a horse.
 - 2.3 Bluebeard is a horse.
 - 2.4 Charlie is Bluebeard's offspring.
 - 2.5 Offspring and parent are inverse relations.
 - 2.6 Every mammal has a parent.
3. จากประโยคตรรกะในข้อ 2 จงตอบคำถามต่อไปนี้โดยใช้วิธีแบ็คเวิร์ดเชนนิ่ง พร้อมทั้งเขียนแผนผังในการพิสูจน์ด้วย

$$\exists h \text{ Horse}(h)$$
4. จากประโยค "Horses are animals." มีประโยคที่ตามมาคือ "The head of a horse is the head of an animal."

จงแสดงว่าประโยคหลังนี้เป็นการสรุปความที่เป็นจริง โดย

 - 4.1 แปลงประโยคให้เป็นตรรกศาสตร์อันดับหนึ่ง กำหนดให้ใช้เพรดิเคตเหล่านี้

$$\text{HeadOf}(h, x) \text{ หมายถึง "h is the head of x"}$$

$$\text{Horse}(x) \text{ และ } \text{Animal}(x)$$
 - 4.2 แปลงส่วนผลสรุปในข้อ 4.1 เป็นนิเสธ และเปลี่ยนส่วน Premise และผลสรุปนิเสธให้เป็นรูปแบบ CNF
 - 4.3 พิสูจน์ด้วยวิธีวิธีโซลูชันว่าผลสรุปดังกล่าวเป็นความจริง

5. “เรื่องของพี่น้องเอลริค” กำหนดข้อความต่อไปนี้

Edward's last name is Elric.

Edward and Alfonse are brothers.

Brothers have the same last name.

จงแทนข้อความทั้งหมดให้เป็นประโยคตรรกศาสตร์อันดับหนึ่ง แล้วใช้กระบวนการทางตรรกะและอัลกอริทึมทุกวิธีในบทนี้ พิสูจน์ว่า

“Alfonse's last name is Elric.”

6. จาก “Deathnote World”

ในโลกแห่งนี้มีความลับที่คนทั้งหลายไม่รู้ นั่นคือ การที่มียมทูตอยู่จำนวนหนึ่ง ยมทูตจะมี *Deathnote* ของตัวเองคนละเล่มสำหรับบันทึกความตายของมนุษย์. . .

6.1 จงเขียนประโยคตรรกะแทนกฎต่อไปนี้

“คนผู้ใดที่สัมผัส *Deathnote* จะสามารถมองเห็นยมทูต (Death) ผู้เป็นเจ้าของ *Deathnote* เล่มนั้น”

6.2 จงเพิ่มข้อเท็จจริงเหล่านี้ลงไปในฐานะความรู้

“ไลท์ (Light) เป็นคนที่เก็บ *deathnote* ของยมทูตที่ชื่อ ลุค (Luke) ได้”

6.3 จงพิสูจน์ว่า “ไลท์มองเห็นลุค” โดยใช้กระบวนการตรรกะและอัลกอริทึมทุกวิธีในบทนี้ และสามารถเพิ่มสัจพจน์อื่นที่จำเป็น เพื่อให้สามารถพิสูจน์ได้ว่าข้อนี้เป็นความจริง