First Order Logic

Chastine Fatichah
Departemen Teknik Informatika
Mei 2023

Capaian Pembelajaran Matakuliah

Mahasiswa mampu menjelaskan, merancang, dan menerapkan *knowledge-based intelligent agent* dengan merepresentasikan *knowledge base* menjadi *propositional logic* atau *first order logic* serta memanfaatkan algoritma *resolution, forward* dan *backward chaining* untuk melakukan proses inferensi

Pokok Bahasan

- *First Order Logic* (FOL)
- *Inference* FOL
  - *Forward chaining*
  - *Backward chaining*
  - *Resolution*

First-order logic (FOL)

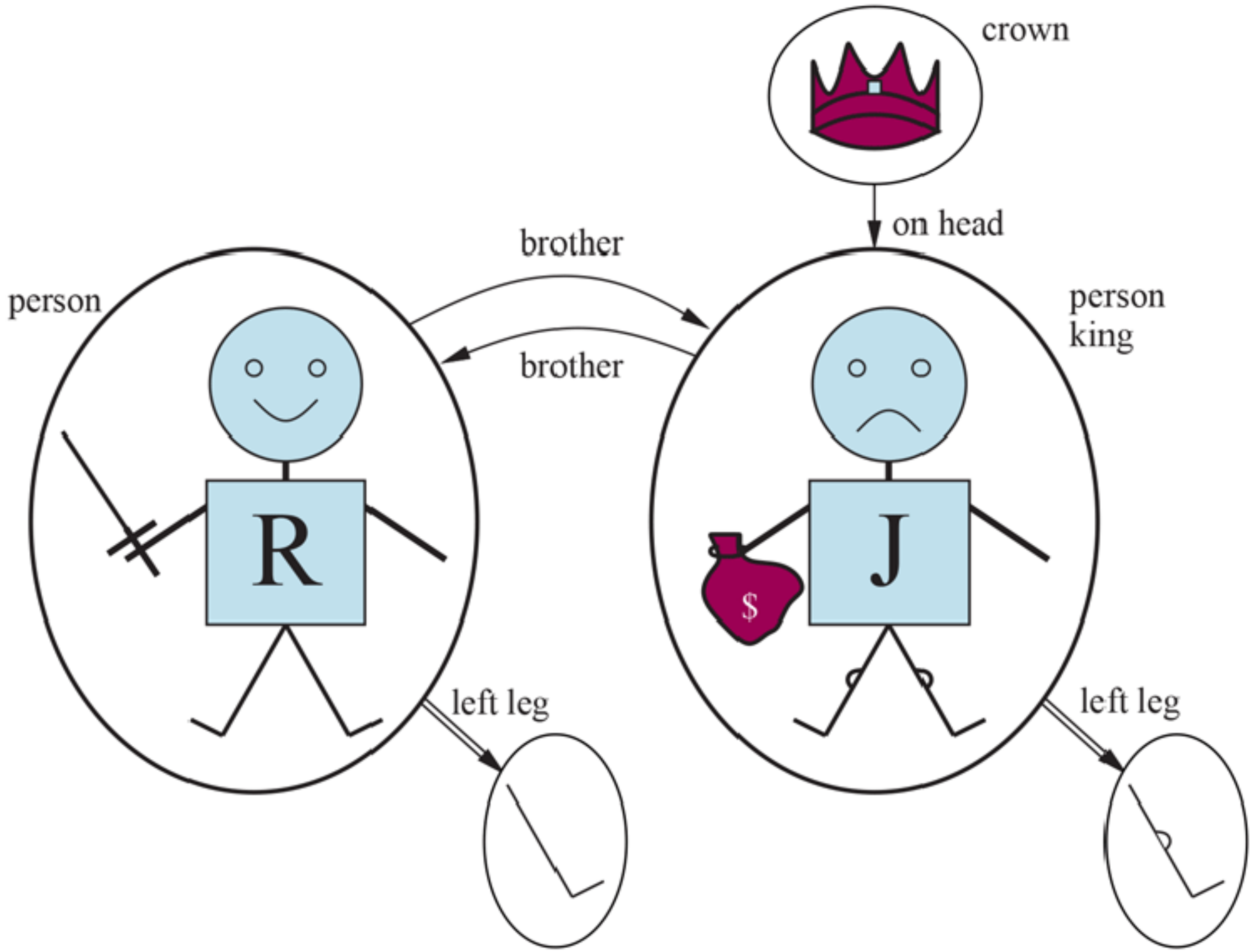
- Logika propositional mengasumsikan dunia dengan fakta-fakta

- *First-order logic* (seperti natural language) mengasumsikan dunia berisi
  - Objects: people, houses, numbers, colors, baseball games, wars, …
  - Relations: red, round, prime, brother of, bigger than, part of, comes between, …
  - Functions: father of, best friend, one more than, plus, …

Syntax FOL

- Constants KingJohn, 2, NUS,...
- Predicates Brother, >,...
- Functions Sqrt, LeftLegOf,...
- Variables x, y, a, b,...
- Connectives $\neg, \Rightarrow, \wedge, \vee, \Leftrightarrow$
- Equality $=$
- Quantifiers $\forall, \exists$

Contoh Model FOL



A model containing five objects, two binary relations (brother and on-head), three unary relations (person, king, and crown), and one unary function (left-leg).

Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Syntax FOL

$$
\begin{array}{rcl}
\textit{Sentence} & \rightarrow & \textit{AtomicSentence} \mid \textit{ComplexSentence} \\
\textit{AtomicSentence} & \rightarrow & \textit{Predicate} \mid \textit{Predicate}(\textit{Term}, \ldots) \mid \textit{Term} = \textit{Term} \\
\textit{ComplexSentence} & \rightarrow & (\ \textit{Sentence}\ ) \\
& \mid & \neg\ \textit{Sentence} \\
& \mid & \textit{Sentence} \land \textit{Sentence} \\
& \mid & \textit{Sentence} \lor \textit{Sentence} \\
& \mid & \textit{Sentence} \ \Rightarrow\ \textit{Sentence} \\
& \mid & \textit{Sentence} \ \Leftrightarrow\ \textit{Sentence} \\
& \mid & \textit{Quantifier Variable}, \ldots\ \textit{Sentence} \\
\\
\textit{Term} & \rightarrow & \textit{Function}(\textit{Term}, \ldots) \\
& \mid & \textit{Constant} \\
& \mid & \textit{Variable} \\
\\
\textit{Quantifier} & \rightarrow & \forall \mid \exists \\
\textit{Constant} & \rightarrow & A \mid X_1 \mid \textit{John} \mid \cdots \\
\textit{Variable} & \rightarrow & a \mid x \mid s \mid \cdots \\
\textit{Predicate} & \rightarrow & \textit{True} \mid \textit{False} \mid \textit{After} \mid \textit{Loves} \mid \textit{Raining} \mid \cdots \\
\textit{Function} & \rightarrow & \textit{Mother} \mid \textit{LeftLeg} \mid \cdots \\
\textsc{Operator Precedence} & : & \neg, =, \land, \lor, \Rightarrow, \Leftrightarrow
\end{array}
$$

Kalimat (Atom)

Atomic sentence = *predicate* ($term_1$,...,$term_n$)
or $term_1$ = $term_2$

Term = *function* ($term_1$,...,$term_n$)
or *constant* or *variable*

Misal: *Brother(KingJohn,RichardTheLionheart)*

> *(Length(LeftLegOf(Richard)), Length(LeftLegOf(KingJohn)))*

Kalimat (Kompleks)

Kalimat komplek didapatkan dari beberapa kalimat atom dengan menggunakan konektivitas

$$\neg S,\ S_1 \wedge S_2,\ S_1 \vee S_2,\ S_1 \Rightarrow S_2,\ S_1 \Leftrightarrow S_2,$$

Misal: *Sibling(KingJohn,Richard)* $\Rightarrow$ *Sibling(Richard,KingJohn)*

$>(1,2) \vee \leq (1,2)$

$>(1,2) \wedge \neg >(1,2)$

Universal Quantifiers

$\forall$<*variables*> <*sentence*>

Everyone at NUS is smart:
$\forall x\ At(x,NUS) \Rightarrow Smart(x)$

$\forall x$ *P* bernilai true di sebuah model *m* iff *P* bernilai benar dengan *x* di setiap obyek pada model

Ekuivalensi *conjunction* pada *instantiations P*

Contoh

$$\begin{array}{ll} & At(KingJohn, NUS) \Rightarrow Smart(KingJohn) \\ \wedge & At(Richard, NUS) \Rightarrow Smart(Richard) \\ \wedge & At(NUS, NUS) \Rightarrow Smart(NUS) \\ \wedge \ldots & \end{array}$$

Universal Quantifiers

$\Rightarrow$ adalah konektvitas utama dengan $\forall$ (Universal Quantifier)

Kesalahan umum menggunakan $\wedge$ sebagai konektivitas utama dengan $\forall$:

Contoh

$\forall x$ At(x, NUS) $\wedge$ Smart(x)
artinya "Everyone is at NUS and everyone is smart"

Existential Quantifiers

$\exists$<*variables*> <*sentence*>

Someone at NUS is smart:
$\exists x$ At(x,NUS) $\wedge$ Smart(x)

$\exists x\ P$ bernilai benar pada sebuah model *m* iff *P* adalah benar dengan *x* di beberapa obyek pada model

Ekuivalensi *disjunction* pada *instantiations* pada *P*

Contoh

At(KingJohn,NUS) $\wedge$ Smart(KingJohn)
$\vee$ At(Richard,NUS) $\wedge$ Smart(Richard)
$\vee$ At(NUS,NUS) $\wedge$ Smart(NUS)
$\vee$ ...

Existential Quantifiers

$\wedge$ adalah konektivitas utama dengan $\exists$

Kesalahan umum menggunakan $\Rightarrow$ sebagai konektifitas utama dengan $\exists$:

Contoh

$\exists x$ At(x, NUS) $\Rightarrow$ Smart(x)

bernilai benar jika ada seseorang yang tidak di NUS!

Properti pada *Quantifiers*

- $\forall x\ \forall y$ is the same as $\forall y\ \forall x$
- $\exists x\ \exists y$ is the same as $\exists y\ \exists x$

- $\exists x\ \forall y$ is not the same as $\forall y\ \exists x$
- $\exists x\ \forall y$ Loves(x, y)
  - "There is a person who loves everyone in the world"
- $\forall y\ \exists x$ Loves(x, y)
  - "Everyone in the world is loved by at least one person"

Quantifier duality:

- $\forall x$ Likes(x, IceCream) $\neg\exists x\ \neg$Likes(x, IceCream)
- $\exists x$ Likes(x, Broccoli) $\neg\forall x\ \neg$Likes(x, Broccoli)

Equality

- $term_1 = term_2$ bernilai benar dalam interpretasi jika dan hanya jika $term_1$ dan $term_2$ merefer ke obyek yang sama

- Misalnya pendefinisian *Sibling* pada term *Parent*:
  - $\forall x,y\ Sibling(x,y) \Leftrightarrow [\neg(x = y) \land \exists m,f\ \neg(m = f) \land Parent(m,x) \land Parent(f,x) \land Parent(m,y) \land Parent(f,y)]$

Contoh FOL pada Kalimat

- Brothers are siblings
  - $\forall x,y \; Brother(x,y) \Leftrightarrow Sibling(x,y)$
- One's mother is one's female parent
  - $\forall m,c \; Mother(c) = m \Leftrightarrow (Female(m) \wedge Parent(m,c))$
- “Sibling” is symmetric
  - $\forall x,y \; Sibling(x,y) \Leftrightarrow Sibling(y,x)$

Contoh FOL pada Kalimat

- Brothers are siblings
  - $\forall x,y\ Brother(x,y) \Leftrightarrow Sibling(x,y)$
- One's mother is one's female parent
  - $\forall m,c\ Mother(c) = m \Leftrightarrow (Female(m) \wedge Parent(m,c))$
- “Sibling” is symmetric
  - $\forall x,y\ Sibling(x,y) \Leftrightarrow Sibling(y,x)$

Universal Instantiation

Semua kalimat dengan *universal quantifier* ($\forall$) meng-entail semua *instantiation*-nya:

$$\frac{\forall v\ \alpha}{\mathrm{Subst}(\{v/g\}, \alpha)}$$

untuk sembarang variabel ***v*** and ground term ***g***

Contoh

$\forall x\ King(x) \land Greedy(x) \Rightarrow Evil(x)$ meng-entail:

$King(John) \land Greedy(John) \Rightarrow Evil(John)$
$King(Richard) \land Greedy(Richard) \Rightarrow Evil(Richard)$
$King(Father(John)) \land Greedy(Father(John)) \Rightarrow Evil(Father(John))$
...

Sumber: S. Russel, P. Norving, Artificial Inttelligencen: A Modern Approach

Existential Instantiation

Untuk sembarang kalimat α, variabel *v*, dan konstanta *k* yang tidak muncul di *knowledge-base*:

$$\frac{\exists v\ \alpha}{\mathrm{Subst}(\{v/k\}, \alpha)}$$

Contoh

$\exists x\ Crown(x) \wedge OnHead(x, John)$ meng-entail:

$Crown(C_1) \wedge OnHead(C_1, John)$

Dengan syarat C_1 adalah konstanta baru, disebut konstanta ***Skolem***

Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Unification

$$\text{U}_{\text{NIFY}}(p,q) = \theta \text{ where } \text{S}_{\text{UBST}}(\theta,p) = \text{S}_{\text{UBST}}(\theta,q).$$

KB = ∀x Knows(x, Obama)
Query = AskVars(Knows(x, Obama))
Answer =

- UNIFY (Knows(x, Obama), Knows(Steve, Obama)) = {x/Steve}
- UNIFY (Knows(x, Obama), Knows(Bill, y)) = {x/Bill,y/Obama}
- UNIFY (Knows(x, Obama), Knows(Mother(y), y)) = {y/Obama,x/Mother(Obama)}
- UNIFY (Knows(x, Obama), Knows(Elizabeth, x)) = failure.

function UNIFY(x, y, θ=*empty*) **returns** a substitution to make x and y identical, or *failure*
  if θ = *failure* **then return** *failure*
  else if $x = y$ **then return** θ
  else if VARIABLE?(x) **then return** UNIFY-VAR(x, y, θ)
  else if VARIABLE?(y) **then return** UNIFY-VAR(y, x, θ)
  else if COMPOUND?(x) **and** COMPOUND?(y) **then**
    return UNIFY(ARGS(x), ARGS(y), UNIFY(OP(x), OP(y), θ))
  else if LIST?(x) **and** LIST?(y) **then**
    return UNIFY(REST(x), REST(y), UNIFY(FIRST(x), FIRST(y), θ))
  else return *failure*

function UNIFY-VAR(*var*, x, θ) **returns** a substitution
  if $\{var/val\} \in \theta$ for some *val* **then return** UNIFY(*val*, x, θ)
  else if $\{x/val\} \in \theta$ for some *val* **then return** UNIFY(*var*, *val*, θ)
  else if OCCUR-CHECK?(*var*, x) **then return** *failure*
  else return add $\{var/x\}$ to θ

Sumber: S. Russel, P. Norving, Artificial Inttelligencen: A Modern Approach

Generalized Modus Ponens (GMP)

- *Generalized Modus Ponens* (GMP) merupakan KBs yang hanya berisi **Horn clauses**
  - *Horn clause* adalah kalimat dalam bentuk:

    `(∀x) (P1(x) ^ P2(x) ^ ... ^ Pn(x)) => Q(x)`

    where there are 0 or more `Pi`'s, and the `Pi`'s and `Q` are positive (i.e., un-negated) literals
  - Deduksi menggunakan GMP berupa KBs yang berisi Horn clauses. Pembuktian dimulai dengan adanya axioms/premises pada KB, kemudian derivasi kalimat baru menggunakan GMP sampai kalimat goal/query. Proses ini disebut prosedur inferensi ***forward chaining*** karena konsep "forward" dari KB ke *goal*.

Forward Chaining FOL dengan GMP

function FOL-FC-ASK(KB, α) **returns** a substitution or *false*
  inputs: KB, the knowledge base, a set of first-order definite clauses
    α, the query, an atomic sentence

  while *true* **do**
    $new \leftarrow \{\}$ // *The set of new sentences inferred on each iteration*
    for each *rule* **in** KB **do**
      $\boxed{(p_1 \wedge \ldots \wedge p_n \Rightarrow q) \leftarrow \text{STANDARDIZE-VARIABLES}(rule)}$
      for each θ such that $\text{SUBST}(\theta, p_1 \wedge \ldots \wedge p_n) = \text{SUBST}(\theta, p'_1 \wedge \ldots \wedge p'_n)$
          for some $p'_1, \ldots, p'_n$ in KB
        $q' \leftarrow \text{SUBST}(\theta, q)$
        if q' does not unify with some sentence already in KB or *new* **then**
          add q' to *new*
          $\phi \leftarrow \text{UNIFY}(q', \alpha)$
          if ϕ is not *failure* **then return** ϕ
    if $new = \{\}$ **then return** *false*
    add *new* to KB

Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Backward Chaining FOL dengan GMP

```
function FOL-BC-ASK(KB, query) returns a generator of substitutions
  return FOL-BC-OR(KB, query, { })

function FOL-BC-OR(KB, goal, θ) returns a substitution
  for each rule in FETCH-RULES-FOR-GOAL(KB, goal) do
    (lhs ⇒ rhs) ← STANDARDIZE-VARIABLES(rule)
    for each θ′ in FOL-BC-AND(KB, lhs, UNIFY(rhs, goal, θ)) do
      yield θ′

function FOL-BC-AND(KB, goals, θ) returns a substitution
  if θ = failure then return
  else if LENGTH(goals) = 0 then yield θ
  else
    first,rest ← FIRST(goals), REST(goals)
    for each θ′ in FOL-BC-OR(KB, SUBST(θ, first), θ) do
      for each θ″ in FOL-BC-AND(KB, rest, θ′) do
        yield θ″
```

Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Contoh Knowledge Base

" 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. "

Buktikan bahwa Colonel West is a criminal !!

Sumber: S. Russel, P. Norving, Artificial Inttelligencen: A Modern Approach

FOL dari KB

... it is a crime for an American to sell weapons to hostile nations:

$$American(x) \wedge Weapon(y) \wedge Sells(x,y,z) \wedge Hostile(z) \Rightarrow Criminal(x)$$

Nono … has some missiles: $\exists x\ Owns(Nono,x) \wedge Missile(x)$:

$$Owns(Nono,M_1) \text{ and } Missile(M_1)$$

… all of its missiles were sold to it by Colonel West

$$Missile(x) \wedge Owns(Nono,x) \Rightarrow Sells(West,x,Nono)$$

Missiles are weapons:

$$Missile(x) \Rightarrow Weapon(x)$$

An enemy of America counts as "hostile“:

$$Enemy(x,America) \Rightarrow Hostile(x)$$

West, who is American …

$$American(West)$$

The country Nono, an enemy of America …

$$Enemy(Nono,America)$$

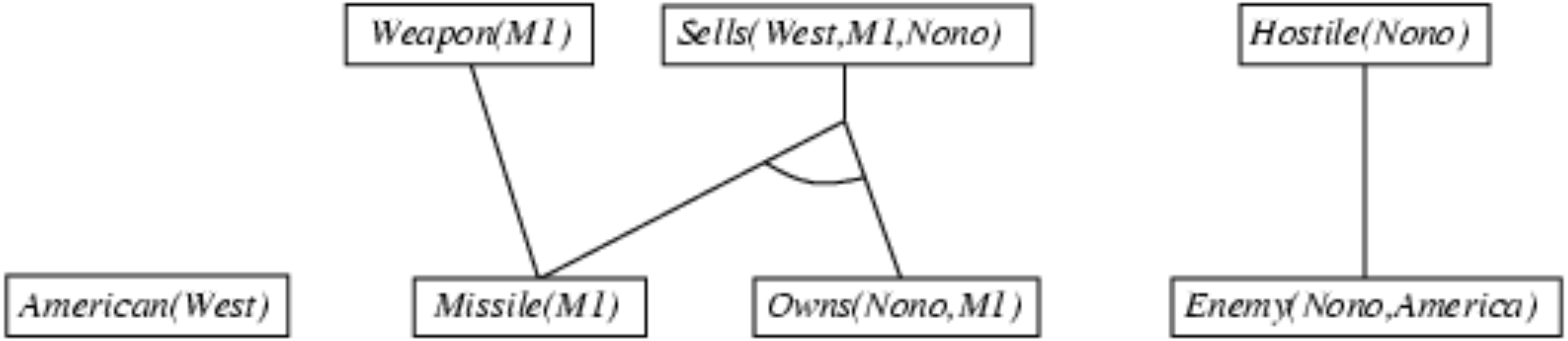
Sumber: S. Russel, P. Norving, Artificial Inttelligencen: A Modern Approach

Contoh Forward Chaining FOL

American(West) and Enemy(Nono,America)

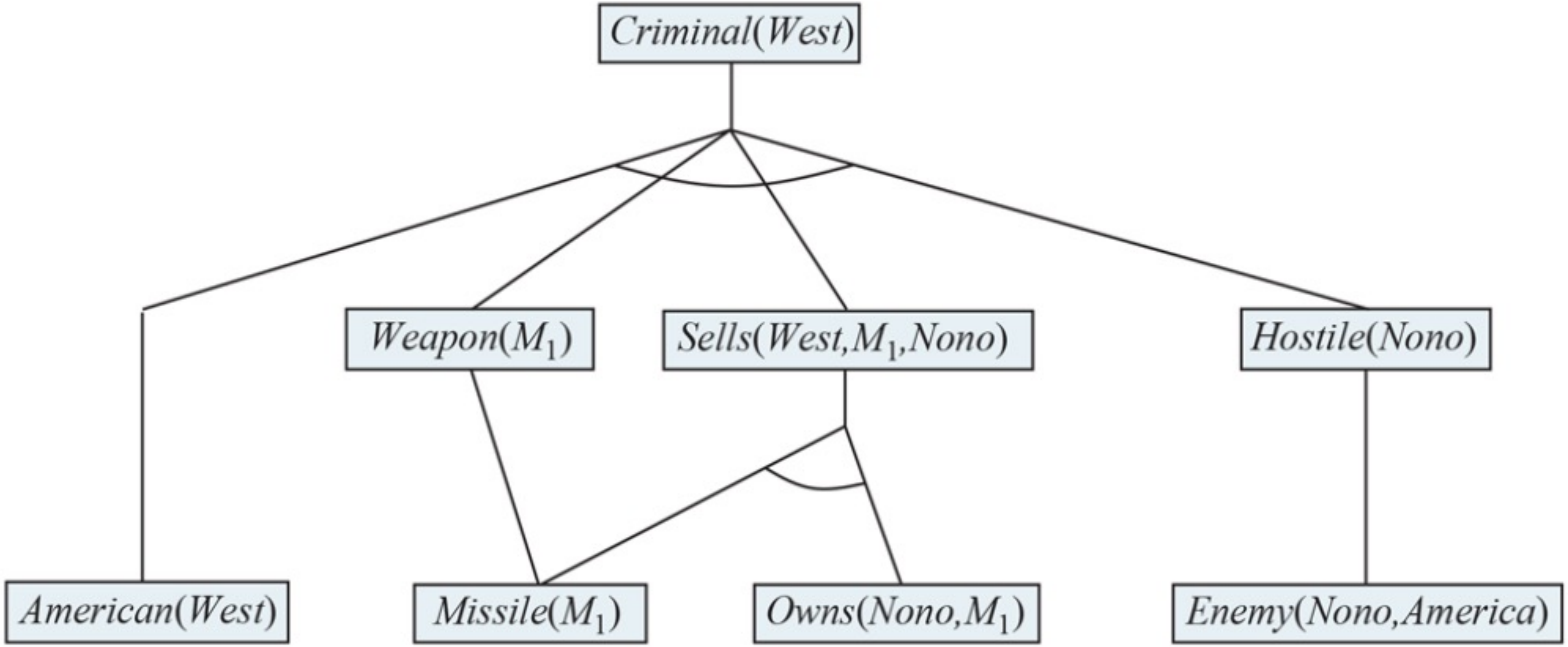
Owns(Nono,M1) and Missile(M1)

Contoh Forward Chaining FOL



Missile(x) ^ Owns(Nono, x) => Sells(West, x, Nono)
Missile(x) => Weapon(x)
Enemy(x, America) => Hostile(x)

Contoh *Forward Chaining* FOL



American(x) ^ Weapon(y) ^ Sells(x, y, z) ^ Hostile(z) => Criminal(x)

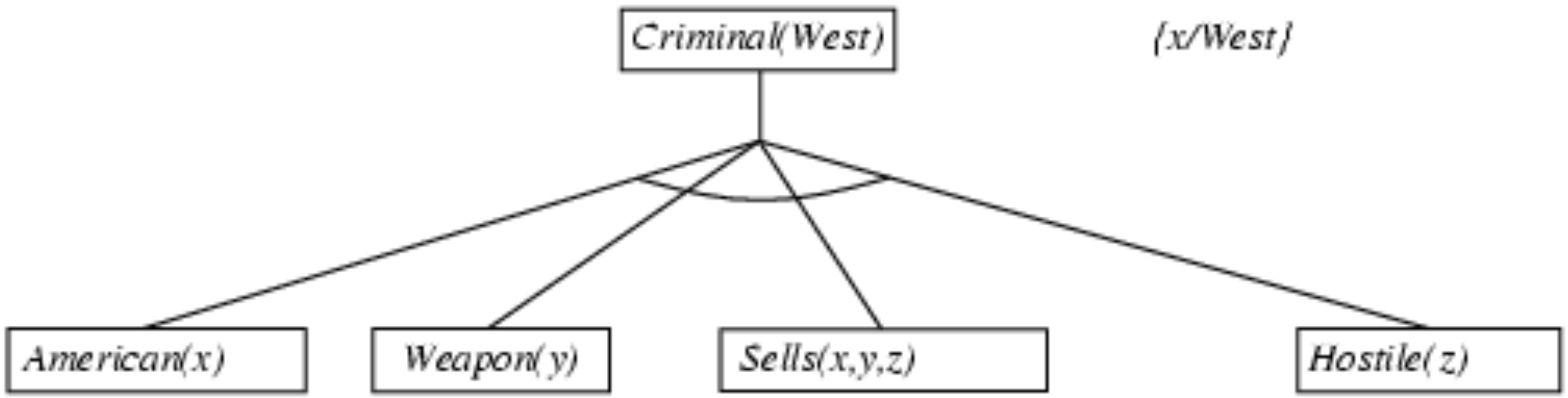
Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Contoh *Backward Chaining* FOL

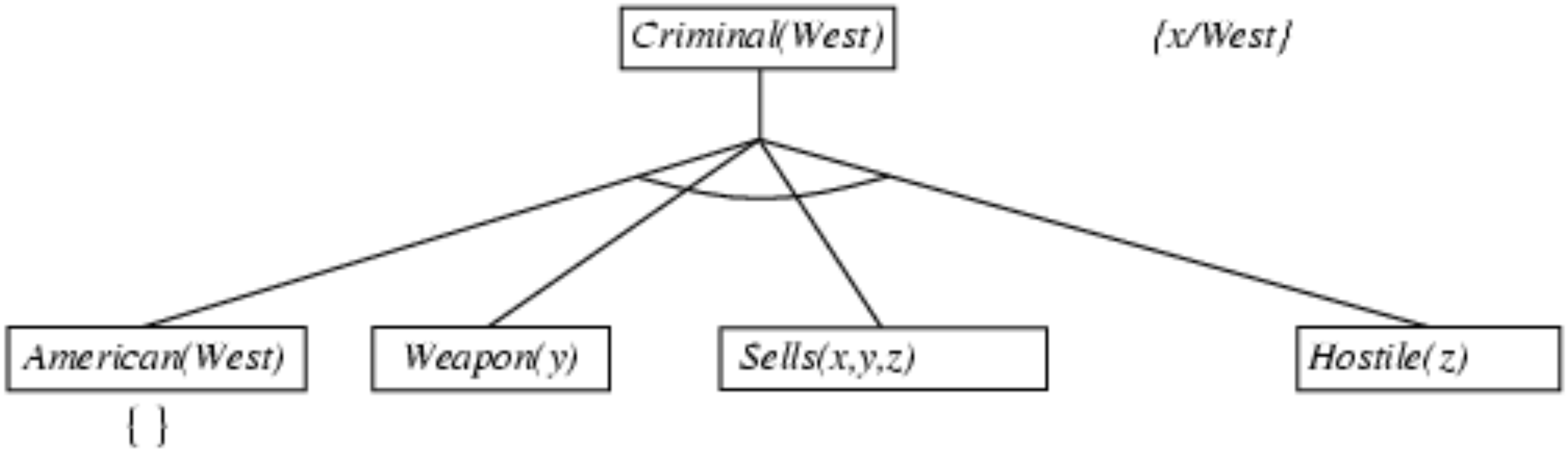
$\boxed{Criminal(West)}$

Contoh *Backward Chaining* FOL

American(x) ^ Weapon(y) ^ Sells(x, y, z) ^ Hostile(z) => Criminal(x)

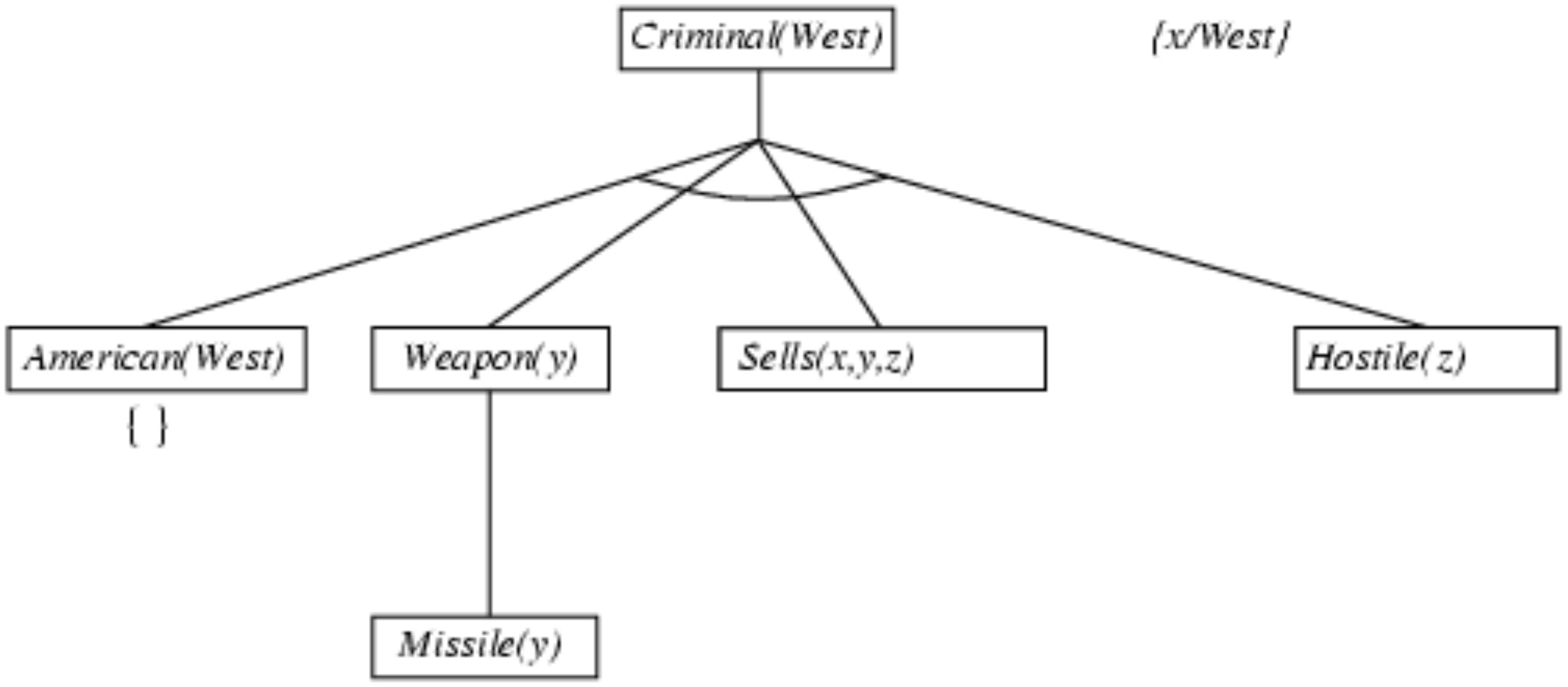


Contoh *Backward Chaining* FOL



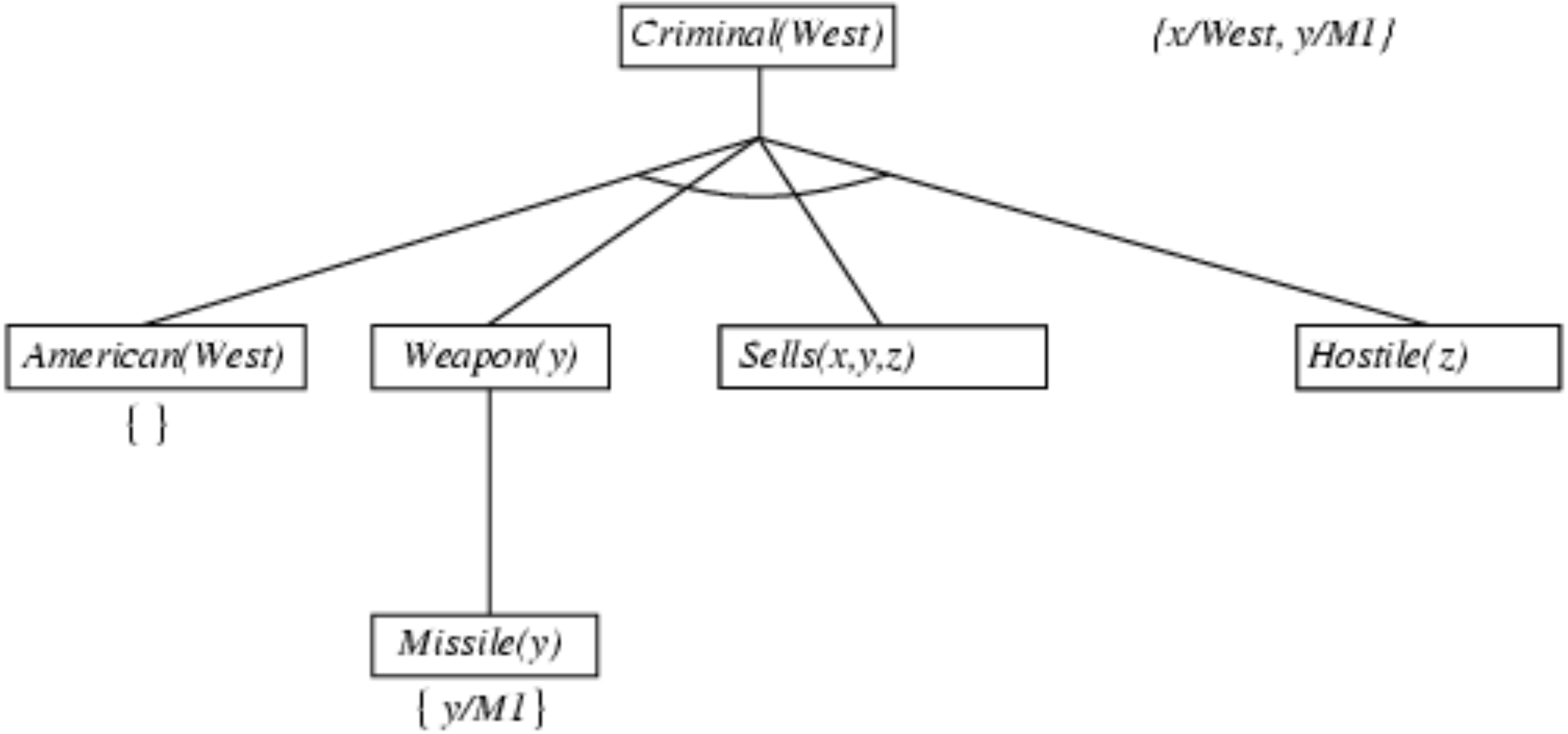
American(West)

Contoh *Backward Chaining* FOL



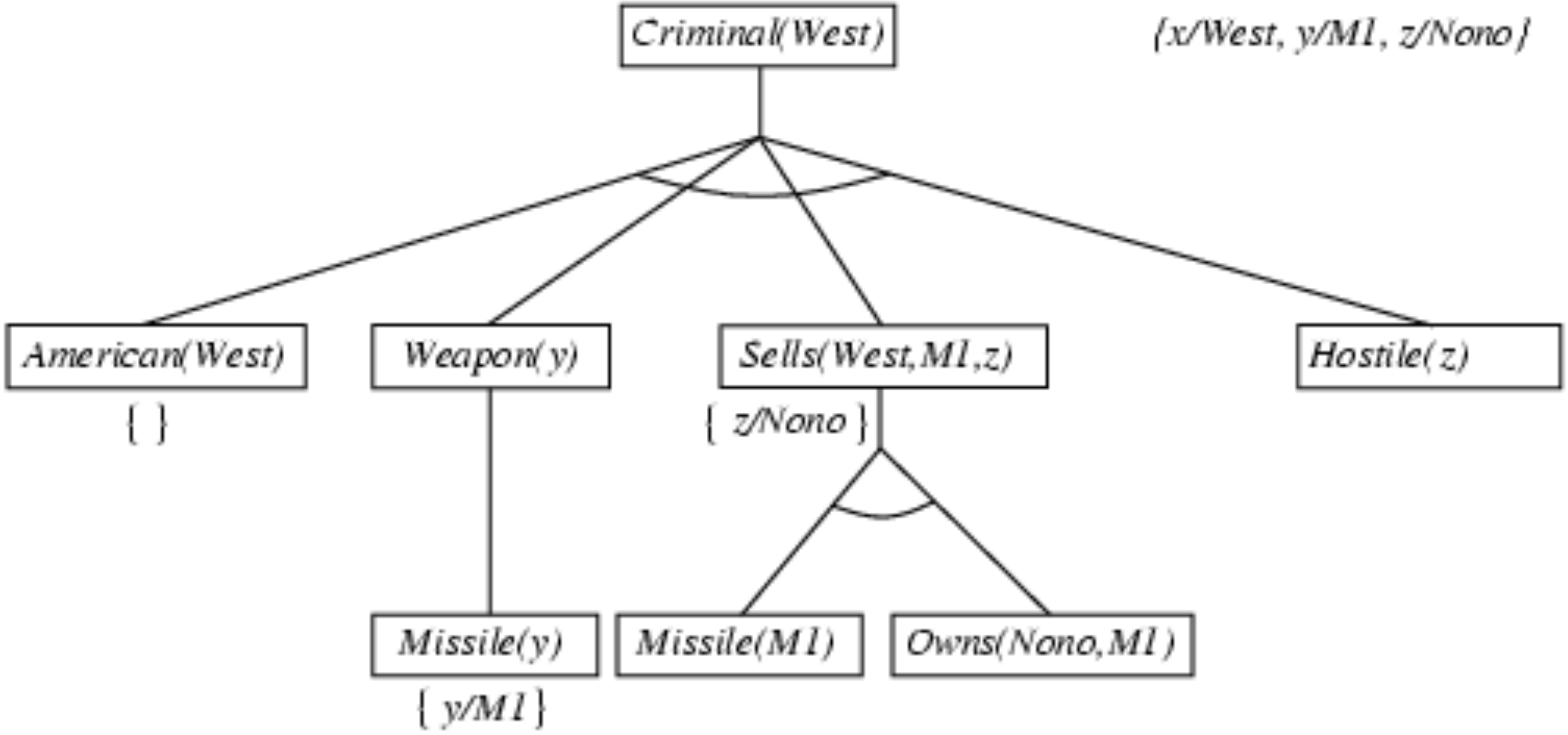
Missile(x) => Weapon(x)

Contoh *Backward Chaining* FOL



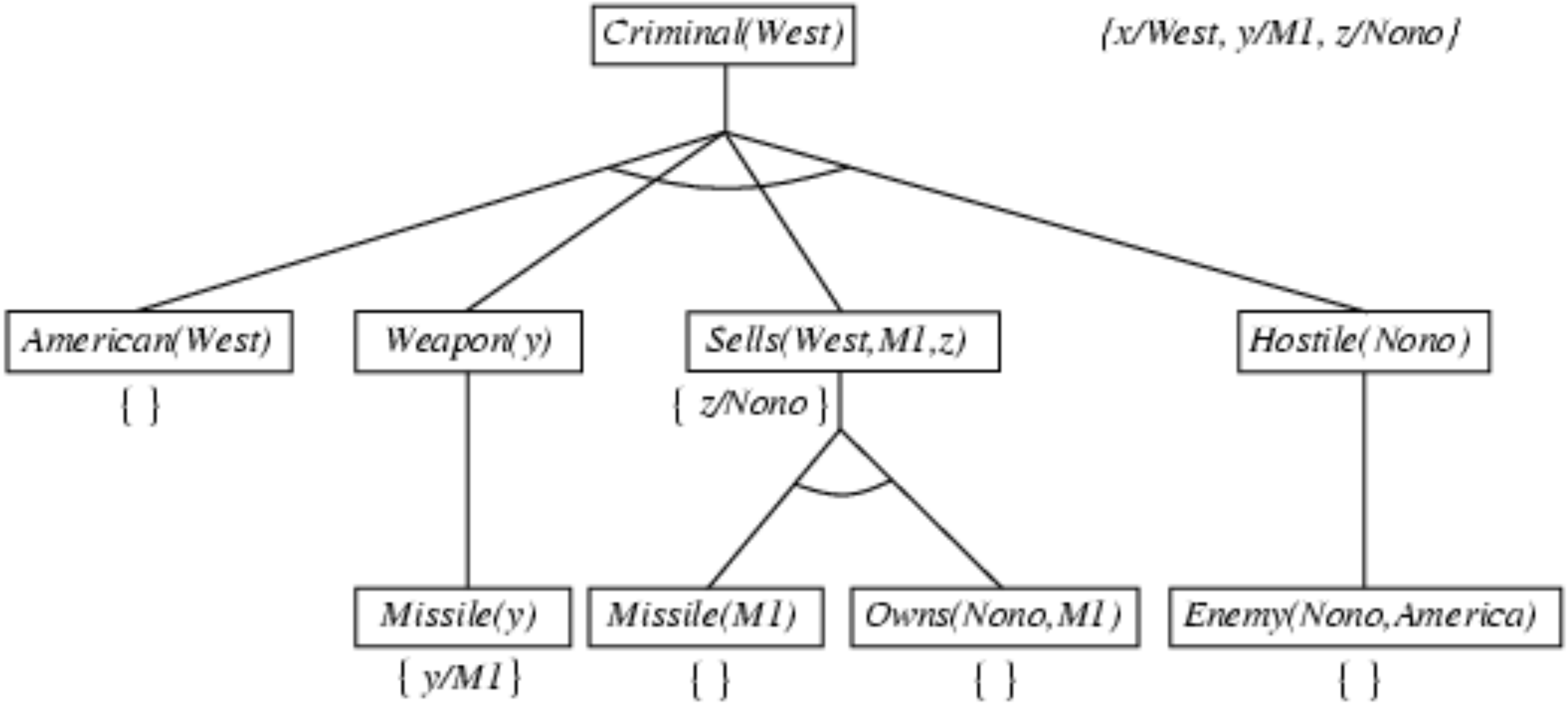
Owns(Nono,M1) and Missile(M1)

Contoh *Backward Chaining* FOL



Missile(x) ^ Owns(Nono, x) => Sells(West, x, Nono)

Contoh *Backward Chaining* FOL

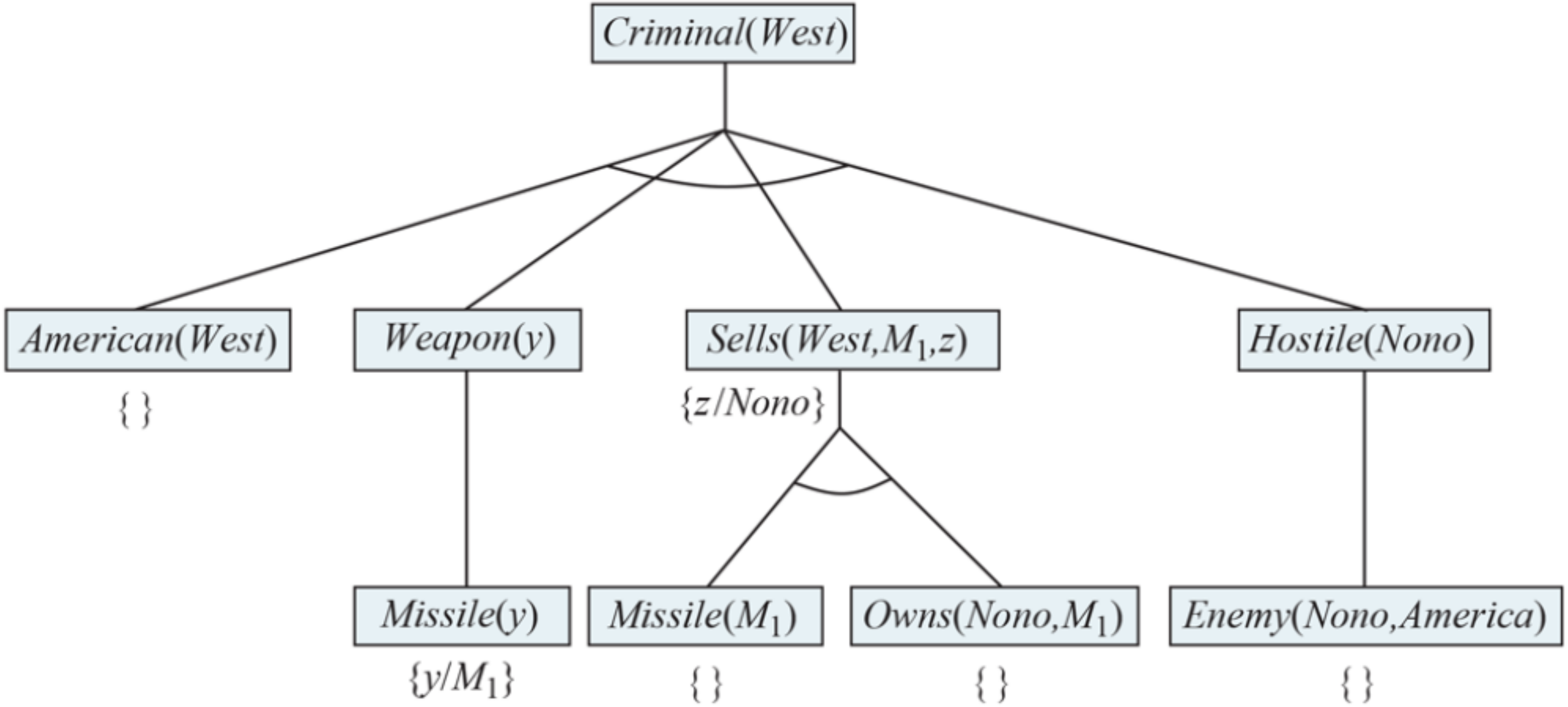


Owns(Nono,M1) and Missile(M1)

Enemy(x, America) => Hostile(x)

American(West) and Enemy(Nono, America)

Contoh *Backward Chaining* FOL



Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Contoh *Resolution* FOL

Dirubah dalam bentuk CNF

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

$Owns(Nono, M_1)$ and $Missile(M_1)$

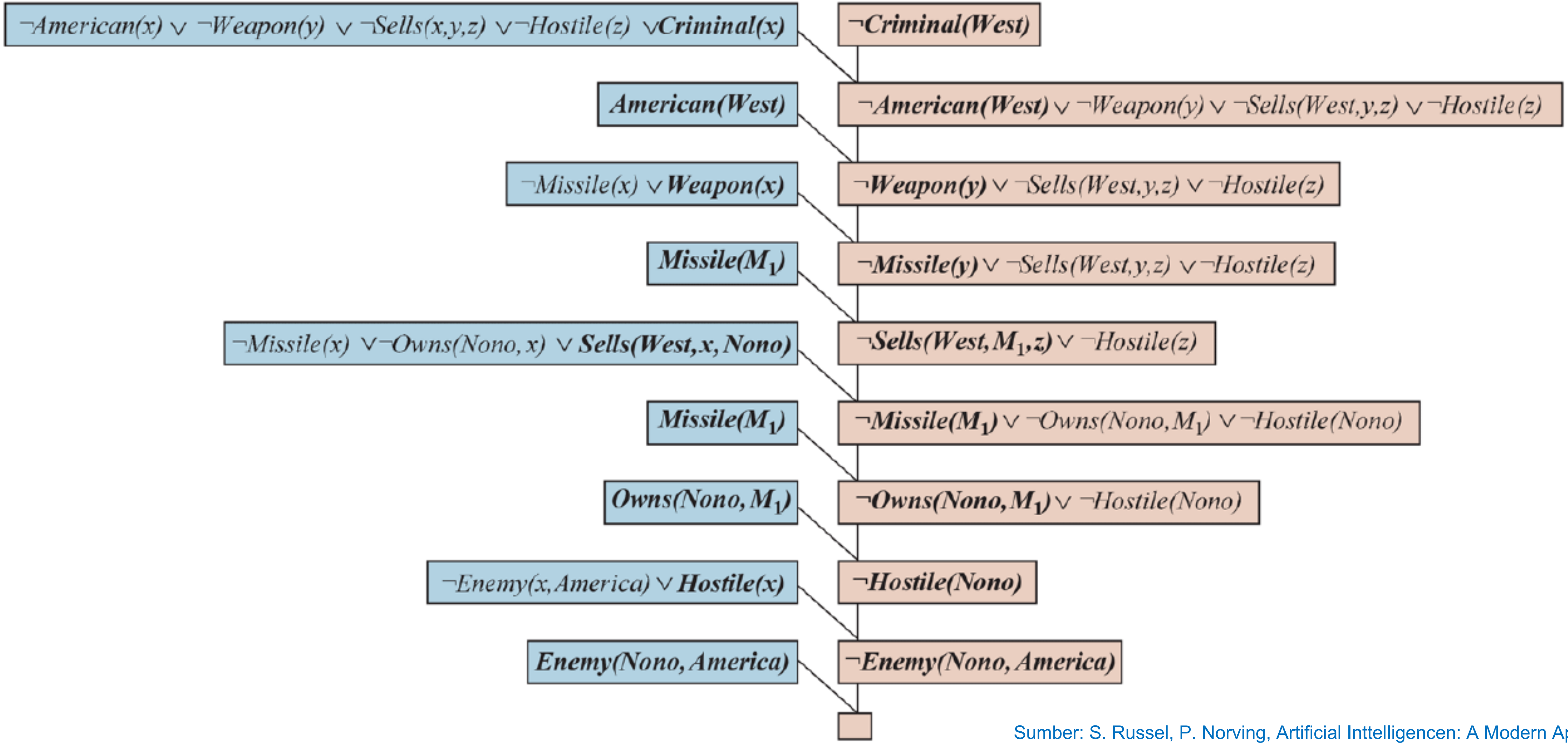
$\neg Missile(x) \vee \neg Owns(Nono, x) \vee Sells(West, x, Nono)$

$\neg Missile(x) \vee Weapon(x)$

$\neg Enemy(x, America) \vee Hostile(x)$

$American(West)$ and $Enemy(Nono, America)$

Contoh *Resolution* FOL



Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Contoh KB

- 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?

Contoh FOL

First, we express the original sentences, some background knowledge, and the negated goal G in first-order logic:

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

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

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

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

E. $Cat(Tuna)$

F. $\forall x\ \ Cat(x) \Rightarrow Animal(x)$

¬G. $\neg Kills(Curiosity, Tuna)$

Sumber: S. Russel, P. Norving, Artificial Intteligencen: A Modern Approach

Merubah FOL ke bentuk CNF

Everyone who loves all animals is loved by someone:

$\forall x\ [\forall y\ Animal(y) \Rightarrow Loves(x,y)] \Rightarrow [\exists y\ Loves(y,x)]$

1. Eliminate biconditionals and implications

$\forall x\ [\neg\forall y\ \neg Animal(y) \vee Loves(x,y)] \vee [\exists y\ Loves(y,x)]$

2. Move $\neg$ inwards: $\neg\forall x\ p \equiv \exists x\ \neg p$, $\neg\exists x\ p \equiv \forall x\ \neg p$

$\forall x\ [\exists y\ \neg(\neg Animal(y) \vee Loves(x,y))] \vee [\exists y\ Loves(y,x)]$

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

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

Merubah FOL ke bentuk CNF

3. Standardize variables: each quantifier should use a different one

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

4. Skolemize: a more general form of existential instantiation.

   Each existential variable is replaced by a Skolem function of the enclosing universally quantified variables:

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

5. Drop universal quantifiers:

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

6. Distribute $\vee$ over $\wedge$:

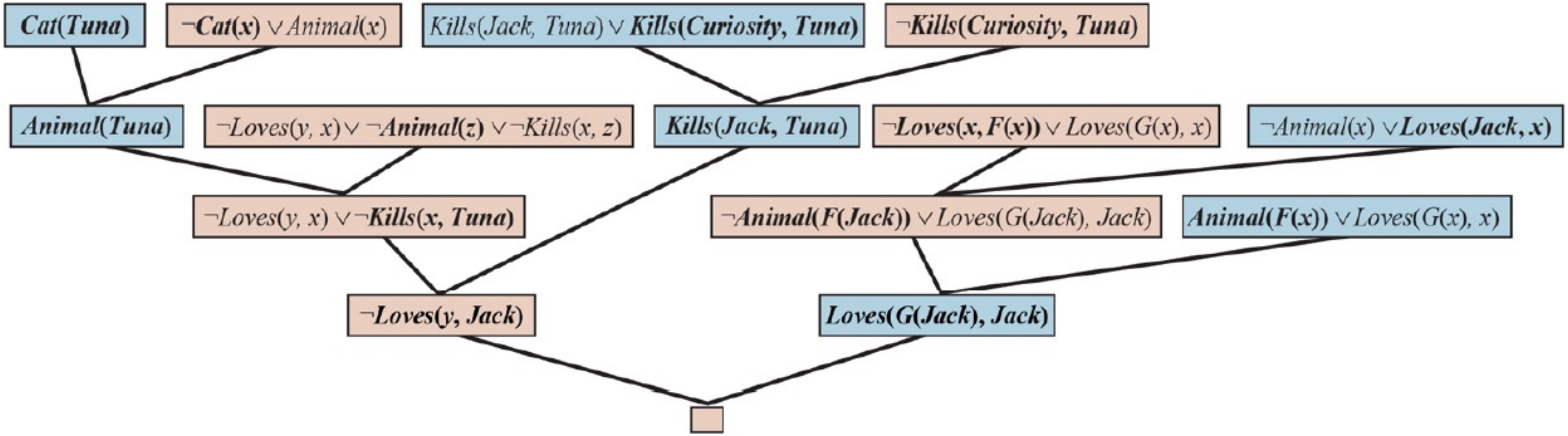
   $[\mathit{Animal}(F(x)) \vee \mathit{Loves}(G(x),x)] \wedge [\neg \mathit{Loves}(x,F(x)) \vee \mathit{Loves}(G(x),x)]$

Hasil Konversi ke CNF

A1.	$Animal(F(x)) \lor Loves(G(x), x)$
A2.	$\neg Loves(x, F(x)) \lor Loves(G(x), x)$
B.	$\neg Animal(y) \lor \neg Kills(x, y) \lor \neg Loves(z, x)$
C.	$\neg Animal(x) \lor Loves(Jack, x)$
D.	$Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna)$
E.	$Cat(Tuna)$
F.	$\neg Cat(x) \lor Animal(x)$
¬G.	$\neg Kills(Curiosity, Tuna)$

Sumber: S. Russel, P. Norving, Artificial Inttelligencen: A Modern Approach

Hasil Konversi ke CNF



A resolution proof that Curiosity killed the cat. Notice the use of factoring in the derivation of the clause $Loves(G(Jack), Jack)$. Notice also in the upper right, the unification of $Loves(x, F(x))$ and $Loves(Jack, x)$ can only succeed after the variables have been standardized apart.

Sumber: S. Russel, P. Norving, Artificial Intelligencen: A Modern Approach

Latihan

- Buat FOL dari kalimat berikut:
  - Every gardener likes the sun
  - All purple mushrooms are poisonous
  - You can fool some of the people all of the time
  - You can fool all of the people some of the time
  - No purple mushroom is poisonous

Latihan

KB = All cats like fish, cats eat everything they like,
and Ziggy is a cat.

Goal query: Does Ziggy eat fish?

a. Buatkan FOL
b. Selesaikan dengan Forward Chaining
c. Selesaikan dengan Backward Chaining
d. Selesaikan dengan Resolution


- TERIMA KASIH -
Teknik Informatika
department of informatics
Fakultas Teknologi Informasi