First-order Logic (FOL)

Definitions

Universal Generalization:

$$\frac{P(c) \text{ for arbitrary } c \in U}{\therefore \forall x P(x)}$$

Existential Instantiation

$$\frac{\exists x P(x)}{\therefore \text{P(c) for some c}}$$

Existential Generalization

$$\frac{\text{P(c) for some element c}}{\therefore \exists x P(x)}$$

Logical Relationships with Quantifiers

- $\forall x[P(x) \land Q(x)] \equiv (\forall xP(x) \land \forall xQ(x))$
- $\exists x[P(x) \lor Q(x)] \equiv (\exists xP(x) \lor \exists xQ(x))$

- $\forall x[P(x) \lor Q(x)]$ *is not* $\equiv (\forall xP(x) \lor \forall xQ(x))$
- $\exists x[P(x) \land Q(x)]$ *is not* $\equiv (\exists xP(x) \land \exists xQ(x))$

A common mistake to avoid

$$\forall x \;\; At(x, Berkeley) \wedge Smart(x)$$

$$\forall x \;\; At(x, Berkeley) \Rightarrow Smart(x)$$

"Everyone at Berkeley is smart"

A common mistake to avoid

Typically, $\Rightarrow$ is the main connective with $\forall$

Common mistake: using $\wedge$ as the main connective with $\forall$:

$$\forall x \;\; At(x, Berkeley) \wedge Smart(x)$$

means "Everyone is at Berkeley and everyone is smart"

Compare with

$$\forall x \;\; At(x, Berkeley) \Rightarrow Smart(x)$$

"Everyone at Berkeley is smart"

A common mistake to avoid

$$\exists x \;\; At(x, Stanford) \Rightarrow Smart(x)$$

$$\exists x \;\; At(x, Stanford) \land Smart(x)$$

"Someone at Stanford is smart"

A common mistake to avoid

Typically, $\wedge$ is the main connective with $\exists$

Common mistake: using $\Rightarrow$ as the main connective with $\exists$:

$$\exists\, x \;\; At(x, Stanford) \Rightarrow Smart(x)$$

is true if there is anyone who is not at Stanford!

Compare with

$$\exists\, x \;\; At(x, Stanford) \wedge Smart(x)$$

"Someone at Stanford is smart"

Proof procedure

Application of inference rules

— simply matches the premise patterns of the rules
to the sentences in the KB and

— adds their suitably instantiated conclusion patterns to the KB

Bob is a buffalo	1.	$Buffalo(Bob)$
Pat is a pig	2.	$Pig(Pat)$
Buffaloes outrun pigs	3.	$\forall x, y Buffalo(x) \wedge Pig(y) \Rightarrow Faster(x, y)$
Bob outruns Pat		

Example Proof

Bob is a buffalo	1. $Buffalo(Bob)$
Pat is a pig	2. $Pig(Pat)$
Buffaloes outrun pigs	3. $\forall x, y \;\; Buffalo(x) \land Pig(y) \;\Rightarrow\; Faster(x, y)$
Bob outruns Pat	
AI 1 & 2	4. $Buffalo(Bob) \land Pig(Pat)$

Example Proof

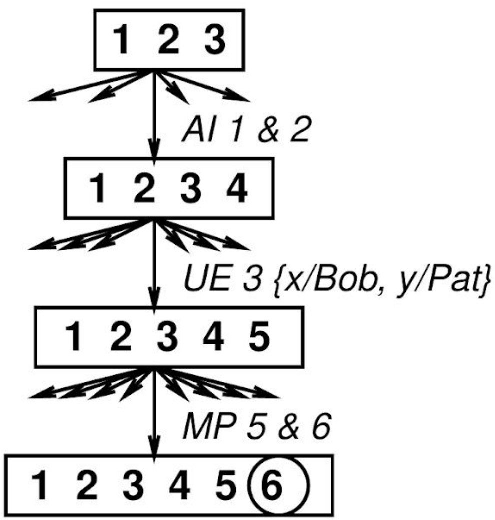
Bob is a buffalo	1. $Buffalo(Bob)$
Pat is a pig	2. $Pig(Pat)$
Buffaloes outrun pigs	3. $\forall x, y \;\; Buffalo(x) \wedge Pig(y) \;\Rightarrow\; Faster(x, y)$
Bob outruns Pat	
UE 3, $\{x/Bob, y/Pat\}$	5. $Buffalo(Bob) \wedge Pig(Pat) \;\Rightarrow\; Faster(Bob, Pat)$

Example Proof

Bob is a buffalo	1. $Buffalo(Bob)$
Pat is a pig	2. $Pig(Pat)$
Buffaloes outrun pigs	3. $\forall x, y \ \ Buffalo(x) \wedge Pig(y) \ \Rightarrow \ Faster(x, y)$
Bob outruns Pat	
MP 6	6. $Faster(Bob, Pat)$

Search with Primitive Inference Rules

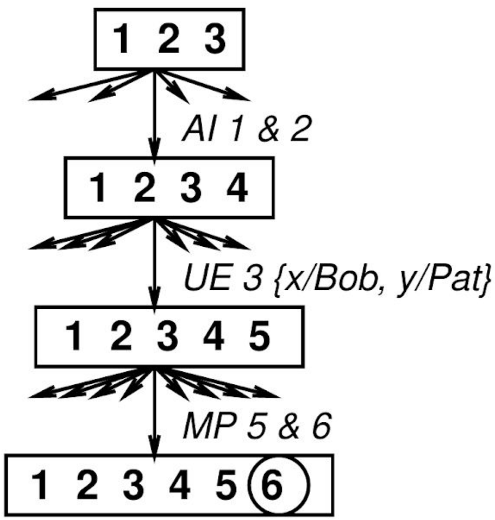
Operators are inference rules
States are sets of sentences
Goal test checks state to see if it contains query sentence



AI, UE, MP is a common inference pattern

Search with Primitive Inference Rules

Operators are inference rules
States are sets of sentences
Goal test checks state to see if it contains query sentence



AI, UE, MP is a common inference pattern

Problem: branching factor huge, esp. for UE

Idea: find a substitution that makes the rule premise match some known facts
$\Rightarrow$ a single, more powerful inference rule

Unification

Example 1:

$$p = Likes(John, x)$$

$$q = Likes(y, IceCream)$$

$$Unify(p, q) = \{John/y, x/IceCream\}$$

Unification

$Unify$:

— takes two atomic sentences p and q and

— returns a substitution in which p and q 'look the same'

— (if not possible, returns $fail$.)

$$Unify(p, q) = \theta \text{ where } Subst(\theta, p) = Subst(\theta, q)$$

Example 1:

$$p = Likes(John, x)$$

$$q = Likes(y, IceCream)$$

$$Unify(p, q) = \{John/y, x/IceCream\}$$

Unification

$Unify$:

— takes two atomic sentences p and q and

— returns a substitution in which p and q 'look the same'

— (if not possible, returns $fail$.)

$$Unify(p, q) = \theta \;\; \text{where} \;\; Subst(\theta, p) = Subst(\theta, q)$$

Example 2:

$$Unify(Likes(John, x), Likes(x, Jane)) = fail$$

Generalized Resolution

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \quad m_1 \vee \cdots \vee m_j \vee \cdots \vee m_k}{\text{SUBST}\,(\theta, l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)}$$

where $\theta = \text{UNIFY}\,(l_i, m_j)$.

Generalized Resolution

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \quad m_1 \vee \cdots \vee m_j \vee \cdots \vee m_k}{\text{SUBST}\ (\theta, l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)}$$

$$\text{where } \theta = \text{UNIFY}\ (l_i, m_j).$$

$$\underbrace{Animal(F(x)) \vee \text{Loves}(G(x), x) \qquad \neg\,\text{Loves}(u, v) \vee \neg\,\text{Kills}(u, v)}_{\text{unifier: } \theta=\{u/G(x), v/x\}}$$

Generalized Resolution

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \quad m_1 \vee \cdots \vee m_j \vee \cdots \vee m_k}{\text{SUBST}\ (\theta, l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)}$$

where $\theta = \text{UNIFY}\ (l_i, m_j)$.

$$\underbrace{\mathit{Animal}(F(x)) \vee \text{Loves}(G(x), x) \qquad \neg\,\text{Loves}(u, v) \vee \neg\,\text{Kills}(u, v)}_{\text{unifier: } \theta = \{u/G(x), v/x\}}$$

Resolvent: $\text{Animal}\ (F(x)) \vee \neg\,\text{Kills}(G(x), x)$

Generalized Modus Ponens

$$\begin{aligned}
\text{Example: } {p_1}' &= \quad Faster(Bob, Pat) \\
{p_2}' &= \quad Faster(Pat, Steve) \\
p_1 \wedge p_2 \Rightarrow q \ &= \quad Faster(x, y) \wedge Faster(y, z) \Rightarrow Faster(x, z) \\
\theta &= \quad \{x/Bob, y/Pat, z/Steve\} \\
Subst(\theta, q) &= \quad Faster(Bob, Steve)
\end{aligned}$$

Generalized Modus Ponens

Idea: a generalization of MP to do in a single blow AI+UE+MP

$$\frac{p_1{}', \;\; p_2{}', \; \ldots \; , \; p_n{}', \;\; (p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q)}{Subst(\theta, q)}$$

$$\begin{aligned}
\text{Example: } p_1{}' = &\quad Faster(Bob, Pat) \\
p_2{}' = &\quad Faster(Pat, Steve) \\
p_1 \wedge p_2 \Rightarrow q \;\; = &\quad Faster(x, y) \wedge Faster(y, z) \Rightarrow Faster(x, z) \\
\theta = &\quad \{x/Bob, y/Pat, z/Steve\} \\
Subst(\theta, q) = &\quad Faster(Bob, Steve)
\end{aligned}$$

Skolemization

- Conversion of sentences FOL to CNF requires skolemization.
- Skolemization: remove existential quantifiers by introducing new function symbols.

Skolemization

Simple case, as in existential elimination (remember?)

$\exists\, x P(x)$ translates into $P(A)$, $\quad A$ constant, not appearing in KB

More complex case: Everyone has a heart

$\forall x\ Person(x) \Rightarrow \exists\, y\ Heart(y) \wedge Has(x, y)$

Skolemization

Simple case, as in existential elimination (remember?)

$\exists\, x P(x)$ translates into $P(A)$, $\quad A$ constant, not appearing in KB

More complex case: Everyone has a heart

$\forall x\; Person(x) \Rightarrow \exists\, y\; Heart(y) \wedge Has(x, y)$

Apply 'simple' existential elimination, replace y with a constant H

$\forall x\; Person(x) \Rightarrow Heart(H) \wedge Has(x, H)$

which means: everyone has the same heart H :—(

Skolemization

Simple case, as in existential elimination (remember?)

$\exists\, xP(x)$ translates into $P(A)$, $\quad A$ constant, not appearing in KB

More complex case: Everyone has a heart

$\forall x\ Person(x) \Rightarrow \exists\, y\ Heart(y) \wedge Has(x, y)$

Apply 'simple' existential elimination, replace y with a constant H

$\forall x\ Person(x) \Rightarrow Heart(H) \wedge Has(x, H)$

which means: everyone has the same heart H :—(

Skolem function: $F(x)$ heart of x, F is a function whose name does not appear elsewhere in the KB

$\forall x\ Person(x) \Rightarrow Heart(F(x)) \wedge Has(x, F(x))$

Skolemization

Original sentence: $\forall x\, Person(x) \Rightarrow \exists\, y\, Heart(y) \wedge Has(x, y)$

Skolemized sentence:
$\forall x\, Person(x) \Rightarrow Heart(F(x)) \wedge Has(x, F(x))$

Skolemization: replace the <u>existentially</u> quantified variable by a Skolem function applied to <u>all</u> the variables <u>universally</u> quantified outside the existentially quantified variable.

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x, y))] \rightarrow [\exists y. \text{Loves}(y, x)])$

(1) Eliminate implications and biconditionals:

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

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x, y))] \rightarrow [\exists y.\, \text{Loves}(y, x)])$

(1) Eliminate implications and biconditionals:

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

(2) Push inwards negations recursively:

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

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x, y))] \rightarrow [\exists y.\, \text{Loves}(y, x)])$

(3) Standardize variables: (Each quantifier should use a different variable)

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x,y))] \rightarrow [\exists y.\, \text{Loves}(y,x)])$

(3) Standardize variables: (Each quantifier should use a different variable)

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

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x, y))] \rightarrow [\exists y.\, \text{Loves}(y, x)])$

(3) Standardize variables:

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

(4) Skolemize:

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

(F(x) : "an animal unloved by x"; G(x) : "someone who loves x")

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x, y))] \rightarrow [\exists y.\, \text{Loves}(y, x)])$

(5) Drop universal quantifiers::

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

Conversion to CNF example

Consider: "Everyone who loves all animals is loved by someone"
$\forall x.([\forall y.(\text{Animal}(y) \rightarrow \text{Loves}(x, y))] \rightarrow [\exists y.\, \text{Loves}(y, x)])$

(5) Drop universal quantifiers::

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

(6) CNF-ize propositionally:

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

Example

KB: 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.

Goal: Prove that Colonel West is a criminal.

Forward Chaining

Example

- it is a crime for an American to sell weapons to hostile nations:
  $\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$

Example

- it is a crime for an American to sell weapons to hostile nations: $\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$

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

Example

- it is a crime for an American to sell weapons to hostile nations:
  $\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$

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

- Nono ... has some missiles
  $\exists x.(Owns(Nono, x) \wedge Missile(x)) \Longrightarrow Owns(Nono, M_1) \wedge Missile(M_1)$
- All of its missiles were sold to it by Colonel West
  $\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$

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

Example

- Missiles are weapons:
  $\forall x.(\mathit{Missile}(x) \rightarrow \mathit{Weapon}(x)) \Longrightarrow \neg \mathit{Missile}(x) \vee \mathit{Weapon}(x)$
- An enemy of America counts as “hostile”:
  $\forall x.(\mathit{Enemy}(x, \mathit{America}) \rightarrow \mathit{Hostile}(x))$

$\Longrightarrow \neg \mathit{Enemy}(x, \mathit{America}) \vee \mathit{Hostile}(x)$

Example

- Missiles are weapons:
  $\forall x.(Missile(x) \rightarrow Weapon(x)) \Longrightarrow \neg Missile(x) \vee Weapon(x)$
- An enemy of America counts as “hostile”:
  $\forall x.(Enemy(x, America) \rightarrow Hostile(x))$

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

- West, who is American ...: $American(West)$
- The country Nono, an enemy of America ...:
  $Enemy(Nono, America)$

Example of Forward Chaining

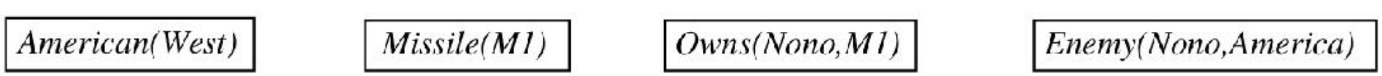
$American(West), Missile(M_1), Owns(Nono, M_1), Enemy(Nono, America)$

$\forall x.(Missile(x) \rightarrow Weapon(x))$

$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$

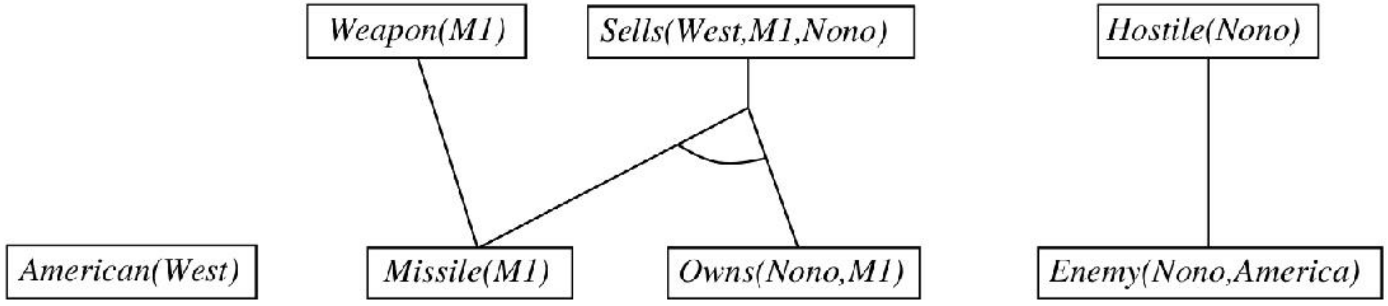
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$

$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$



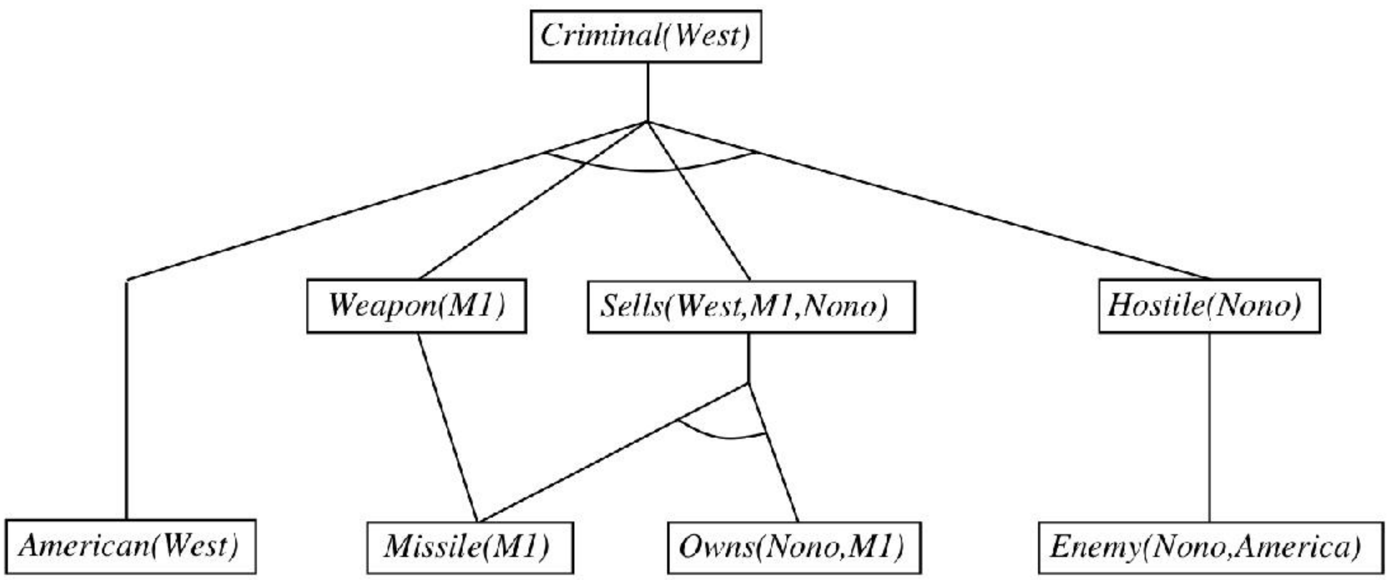
Example of Forward Chaining

$American(West), Missile(M_1), Owns(Nono, M_1), Enemy(Nono, America)$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$



Example of Forward Chaining

$American(West), Missile(M_1), Owns(Nono, M_1), Enemy(Nono, America)$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$



Forward Chaining

- Sound: every inference is just an application of GMP
- Complete (for definite KBs): answers every query entailed by KB
- if $KB \models \alpha$, it always terminates
- if $KB \not\models \alpha$, may not terminate

Backward Chaining

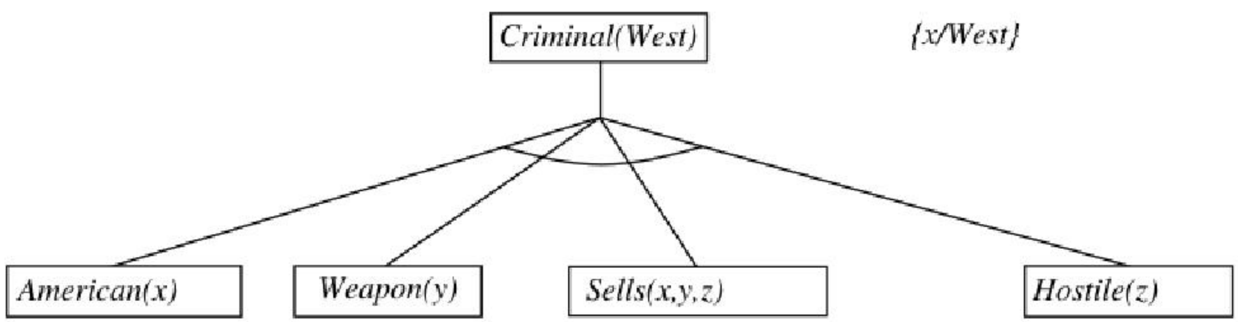
Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$

$\boxed{Criminal(West)}$

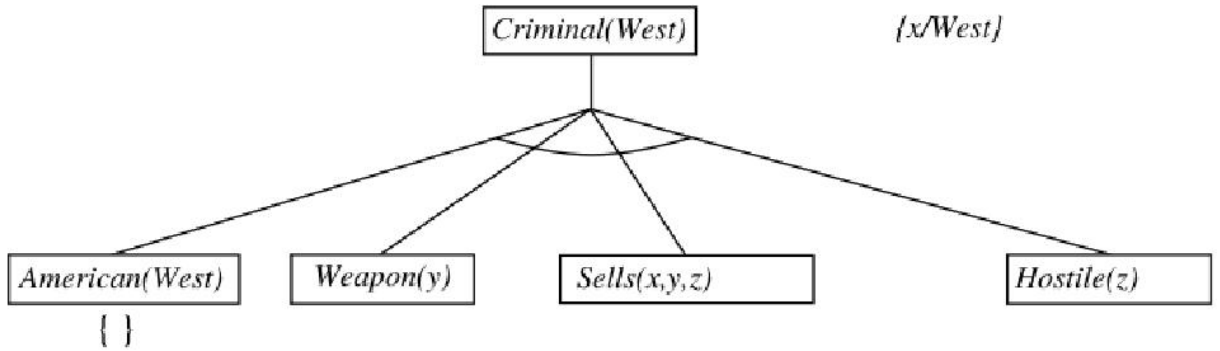
Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \land Weapon(y) \land Hostile(z) \land Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \land Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$



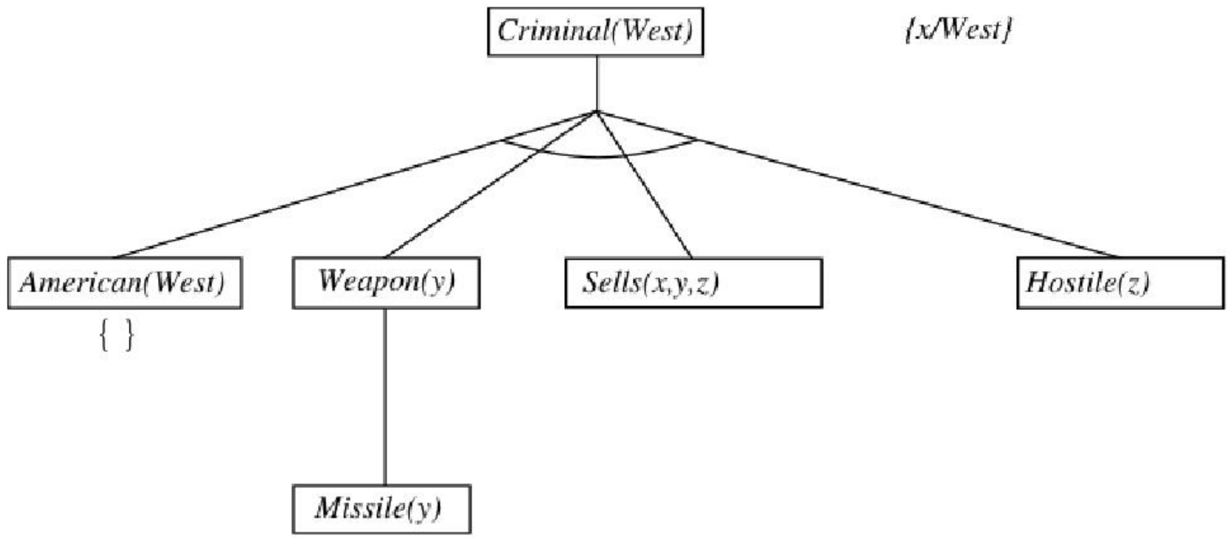
Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$



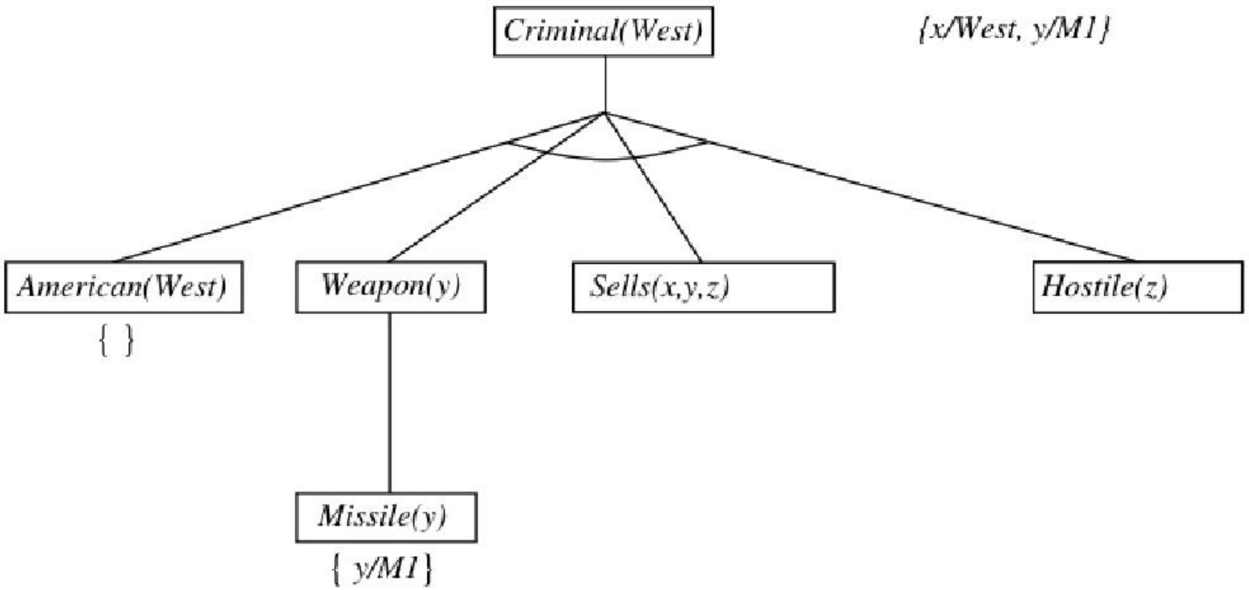
Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$



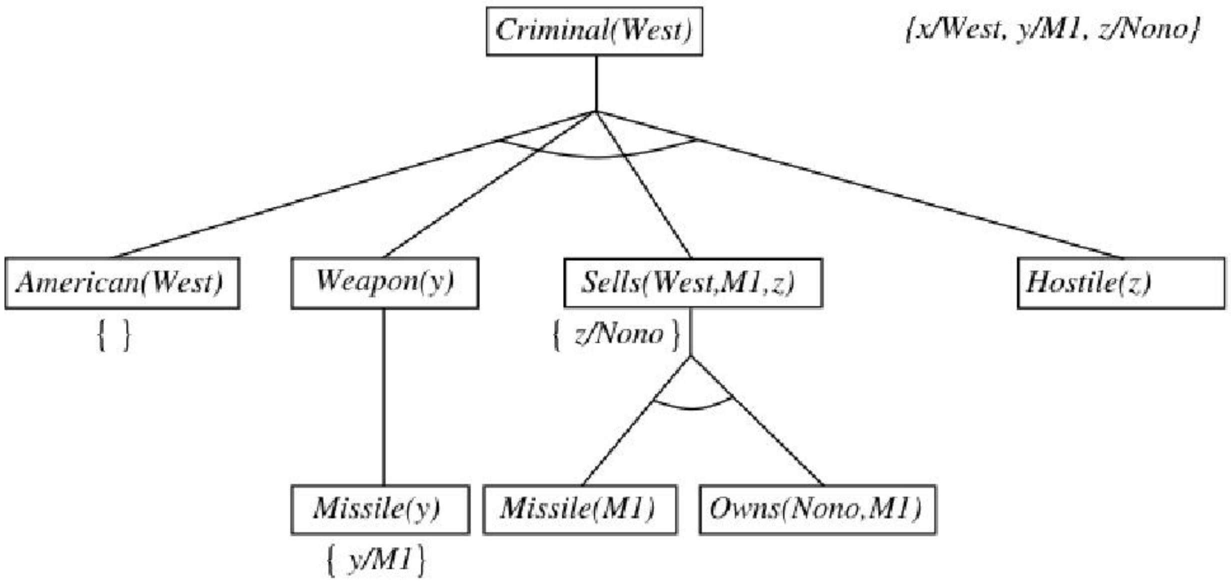
Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$



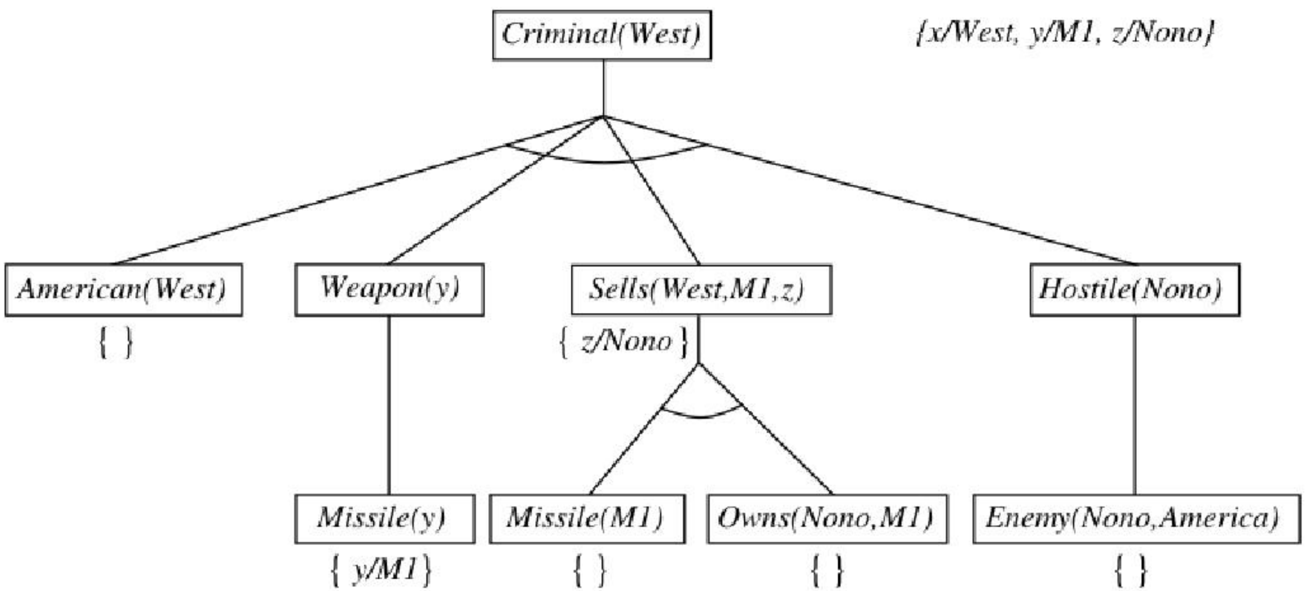
Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$



Example of Backward Chaining

$American(West)$, $Missile(M_1)$, $Owns(Nono, M_1)$, $Enemy(Nono, America)$
$\forall x, y, z.((American(x) \wedge Weapon(y) \wedge Hostile(z) \wedge Sells(x, y, z)) \rightarrow Criminal(x))$
$\forall x.(Missile(x) \rightarrow Weapon(x))$
$\forall x.((Missile(x) \wedge Owns(Nono, x)) \rightarrow Sells(West, x, Nono))$
$\forall x.(Enemy(x, America) \rightarrow Hostile(x))$



Resolution

- Ex: $\dfrac{Man(Socrates) \quad (\neg Man(x) \vee Mortal(x))}{Mortal(Socrates)}$ s.t. $\theta \stackrel{\text{def}}{=} \{x/Socrates\}$
- To prove that $KB \models \alpha$ in FOL:

Resolution

- Ex: $\dfrac{Man(Socrates) \quad (\neg Man(x) \vee Mortal(x))}{Mortal(Socrates)}$ s.t. $\theta \stackrel{\text{def}}{=} \{x/Socrates\}$
- To prove that $KB \models \alpha$ in FOL:
  - convert $KB \wedge \neg\alpha$ to CNF
  - apply repeatedly resolution rule to $CNF(KB \wedge \neg\alpha)$ until
    - the empty clause is generate $\Longrightarrow KB \models \alpha$
    - no more resolution step is applicable $\Longrightarrow KB \not\models \alpha$
    - resource (time, memory) exhausted $\Longrightarrow$??

Resolution Example

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

$\boldsymbol{\neg Criminal(West)}$

Resolution Example

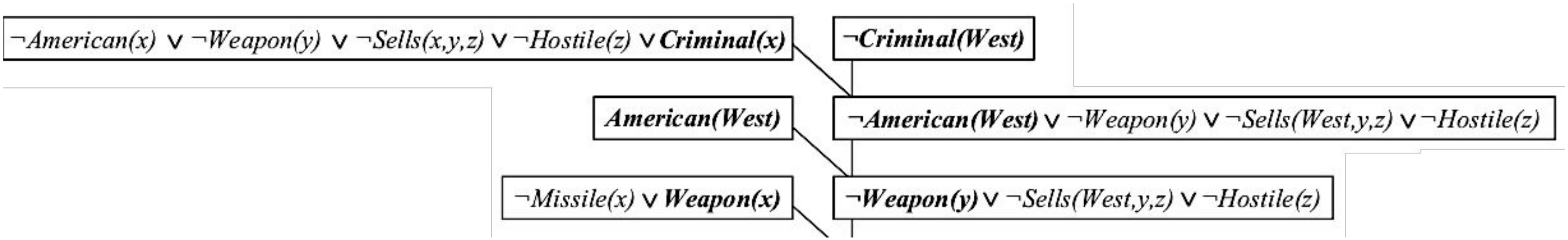
$\neg American(x) \vee \neg Weapon(y) \vee \neg Sells(x,y,z) \vee \neg Hostile(z) \vee \boldsymbol{Criminal(x)}$

$\boldsymbol{\neg Criminal(West)}$

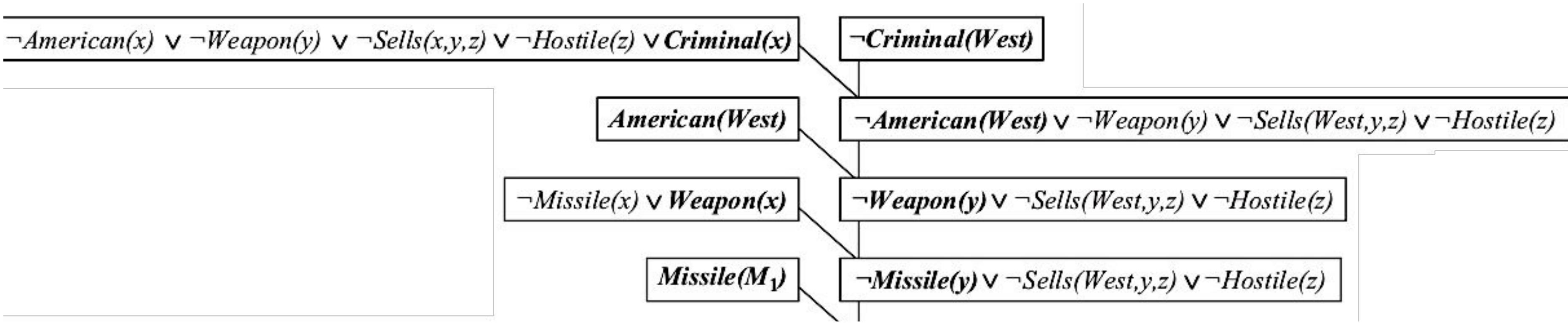
$\boldsymbol{American(West)}$

$\boldsymbol{\neg American(West)} \vee \neg Weapon(y) \vee \neg Sells(West,y,z) \vee \neg Hostile(z)$

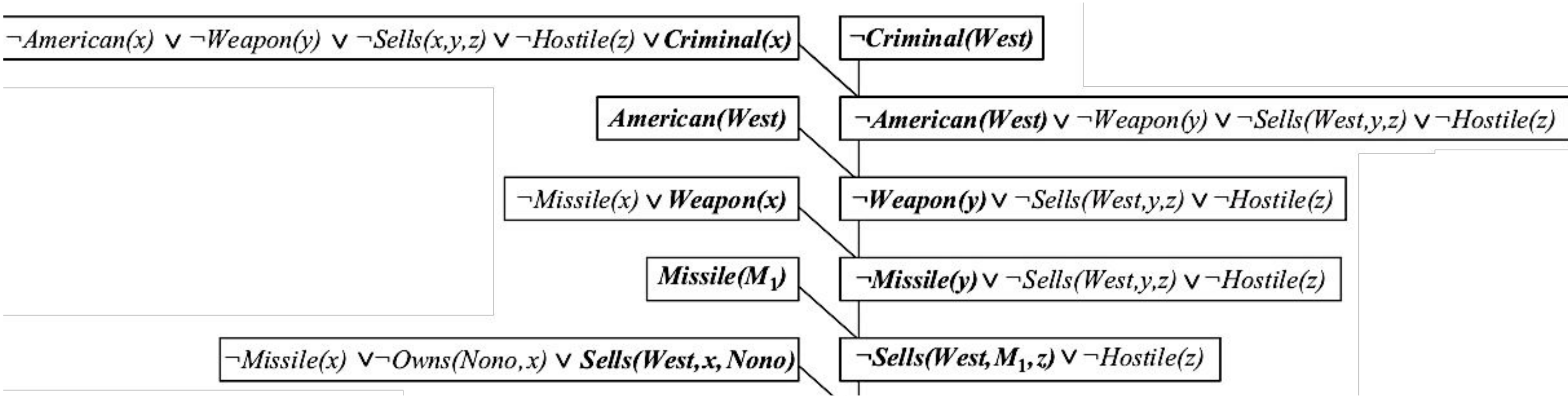
Resolution Example



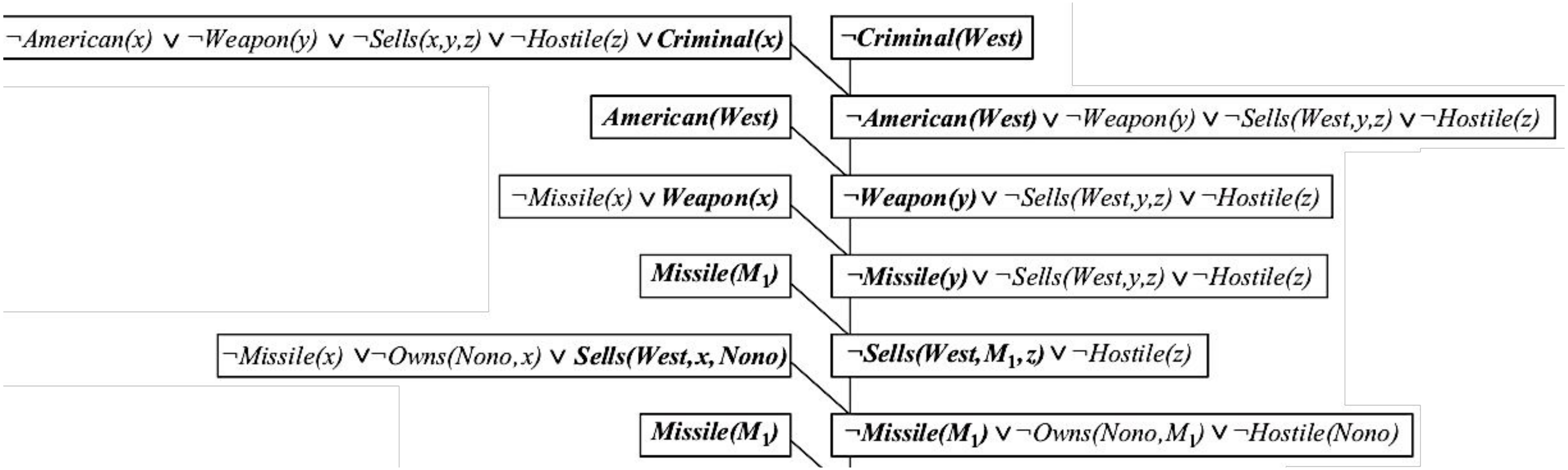
Resolution Example



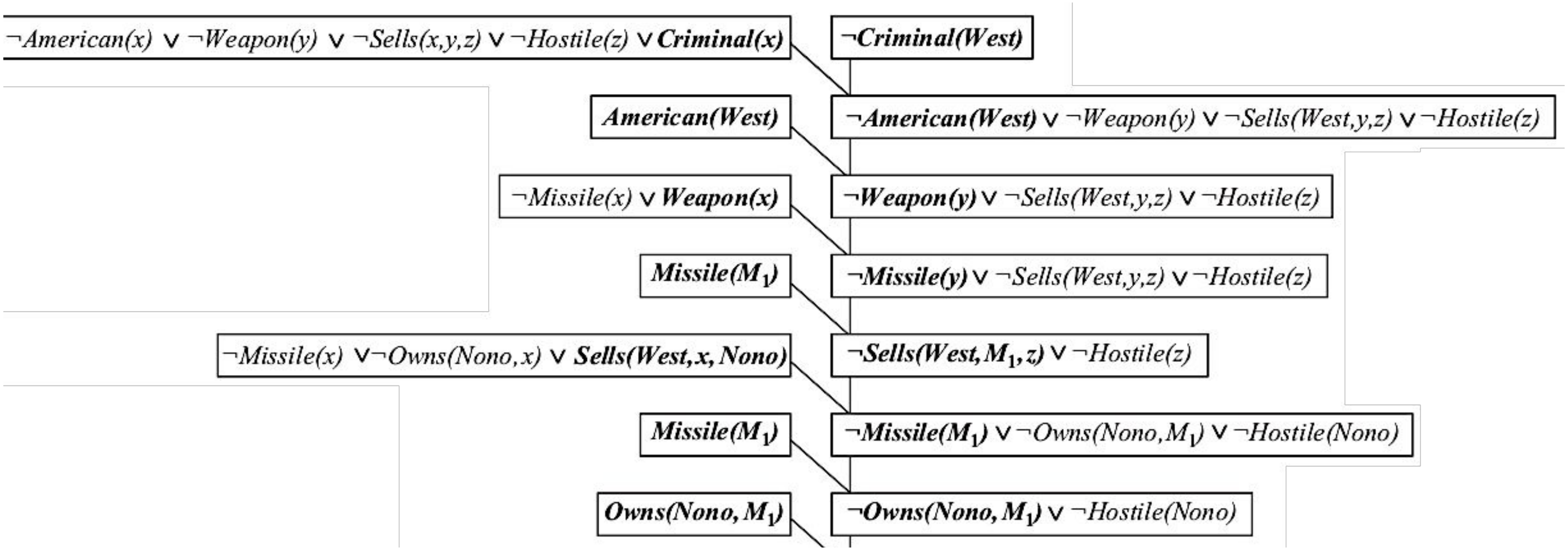
Resolution Example



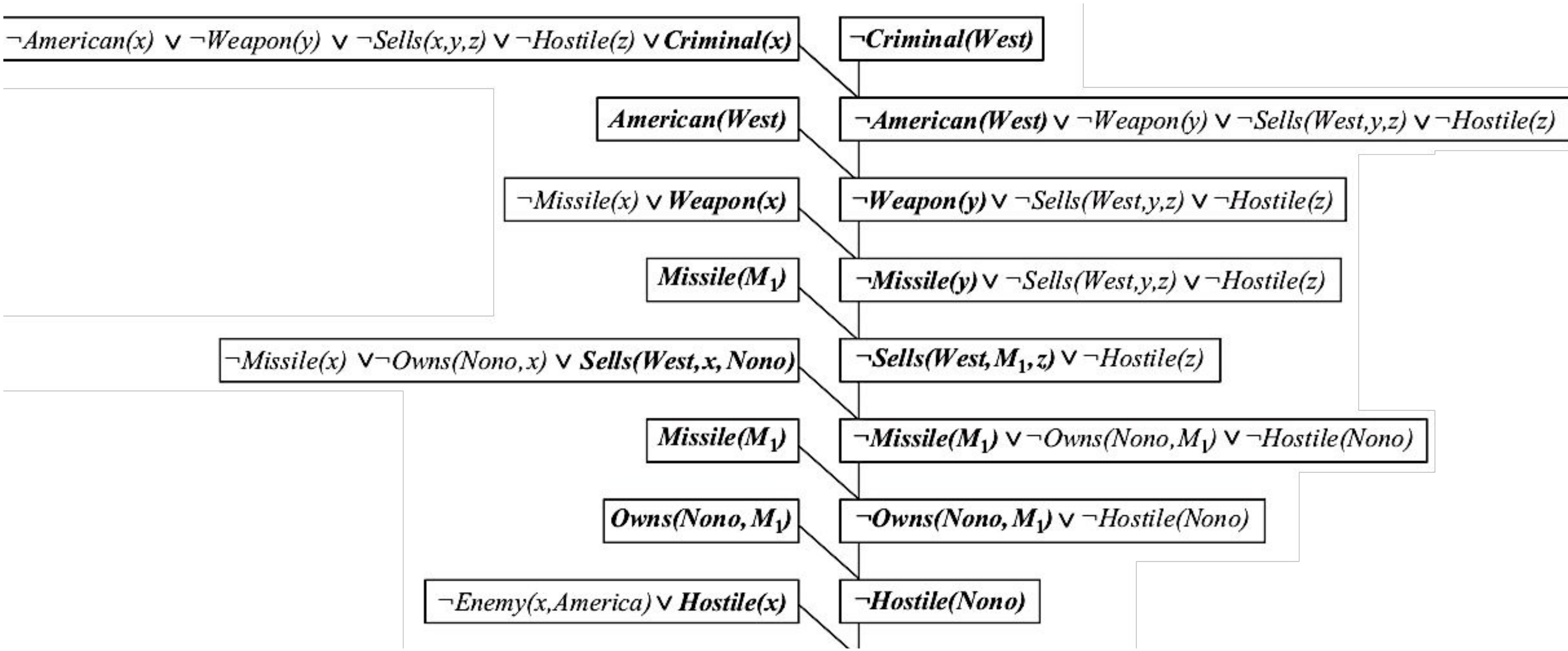
Resolution Example



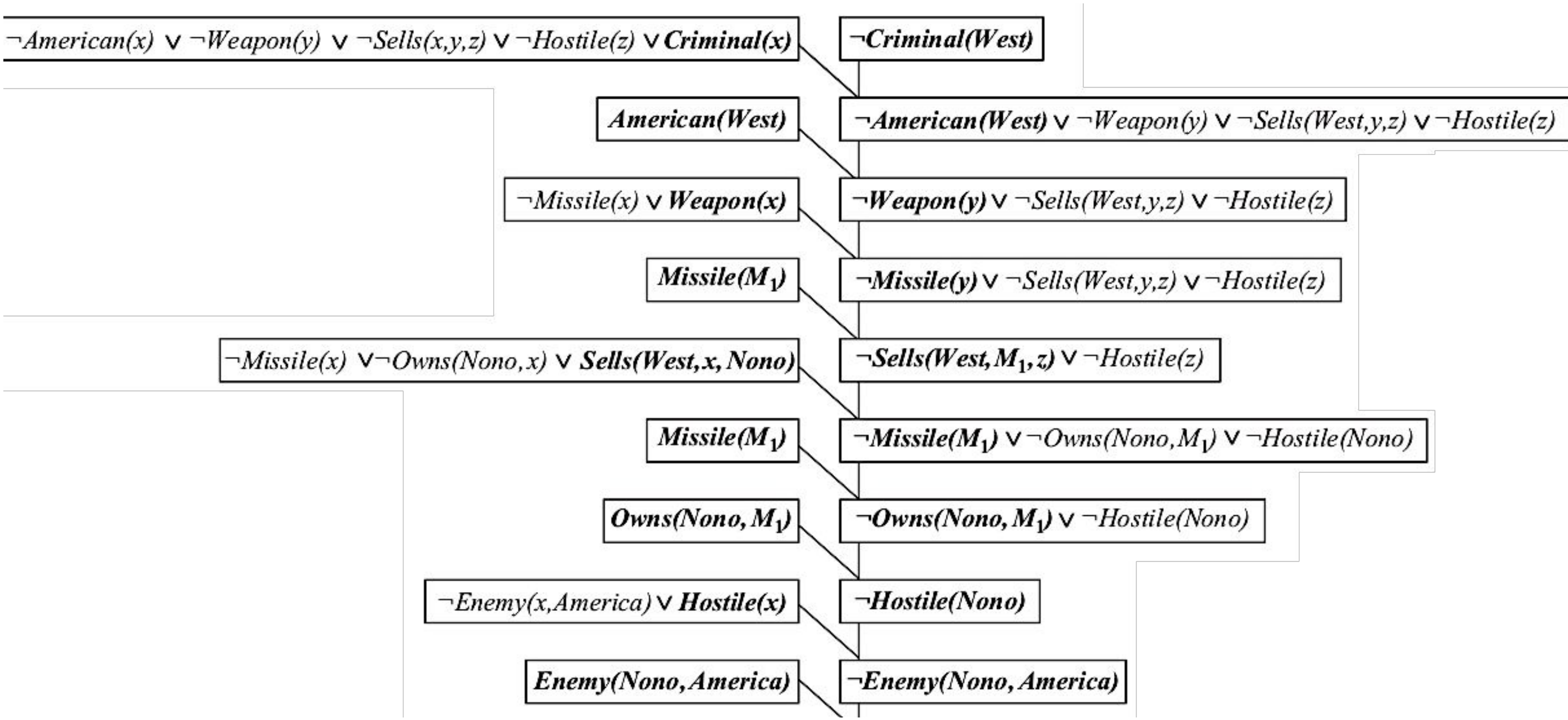
Resolution Example



Resolution Example



Resolution Example



Resolution Example