CAT-2

Foundations of Artificial Intelligence

Unit 3

Propositional Logic.

Propositional Logic	Predicate Logic
1. Each fact is represented by one symbol.	1. representation of the world in terms of object and predicates on object
2. use only connectives	2. use quantifiers as well
3. simple statements	3. complex statements
4. eg. PLATOMAN	4 MAN(PLATO)

Definitions:

(i) Tautology - a sentence that is true under all interpretation

(ii) Sound Inference rule - inference rule does not create any contradictions

a. modus ponens : $A, A \rightarrow B = B$ $\qquad \dfrac{\alpha \Rightarrow \alpha, \beta}{\beta}$

b. and introduction : $A, B = A \wedge B$ $\qquad \dfrac{\alpha_1 \wedge \alpha_2 \wedge \ldots \alpha_n}{\alpha_1}$

c. and elimination : $A \wedge B = A$ $\qquad \dfrac{\alpha_1, \alpha_2 \ldots \alpha_n}{\alpha_1 \wedge \alpha_2 \wedge \ldots \alpha_n}$

d. double negation : $\neg\neg A = A$

e. unit resolution : $A \vee B, \neg B = A$ $\qquad \dfrac{\alpha \vee \beta, \neg \beta}{\alpha}$

f. resolution : $A \vee B, \neg B \vee C = A \vee C$ $\qquad \dfrac{\alpha \vee \beta, \neg B \vee \gamma}{\alpha \vee \gamma}$

(iii) Horn sentences : of the form: $P_1 \wedge P_2 \wedge P_3 \cdots$

(iv) Entailment & Derivation.

Entailment - Q is entailed by the KB only if the conclusion is true in every logically possibly world in which all the premises in KB are true.

$KB \models Q$

Derivation - derive Q from KB - consisting of a valid set of inference steps $KB \vdash Q$

(v) Important properties for inference

if $KB \vdash Q$, then $KB \models Q$

if $KB \models Q$, then $KB \vdash Q$

* Wumpus World Environment

adj to a pit ⇒ breeze

gold ⇒ glitter

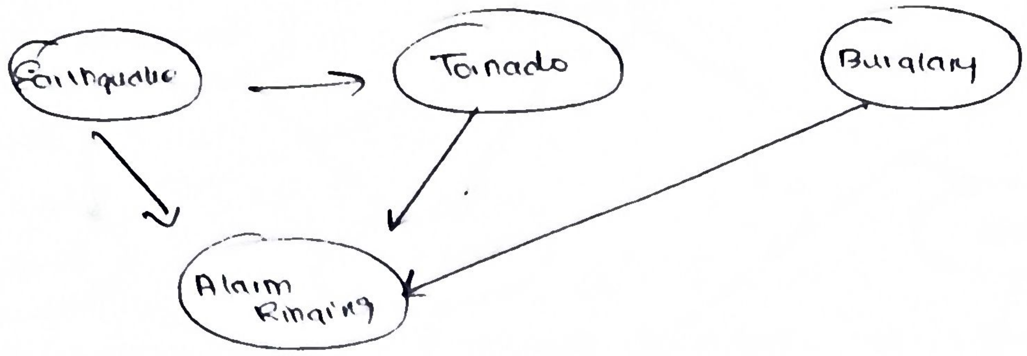
wall ⇒ bump

wumpus ⇒ stench

* Predicate Logic - First Order Logic

FOL models the world in terms of
(i) objects
(ii) properties
(iii) relations
(iv) functions

Example 1



Conditional probability tables

P(E)	P(B)
0.4	0.7

E	P(Tor)
T	0.8
F	0.5

E	B	Tor	P(A)
T	T	T	1.0
T	T	F	0.9
T	F	T	0.95
T	F	F	0.85
F	T	T	0.89
F	T	F	0.7
T	F	T	0.87
F	F	F	0.3

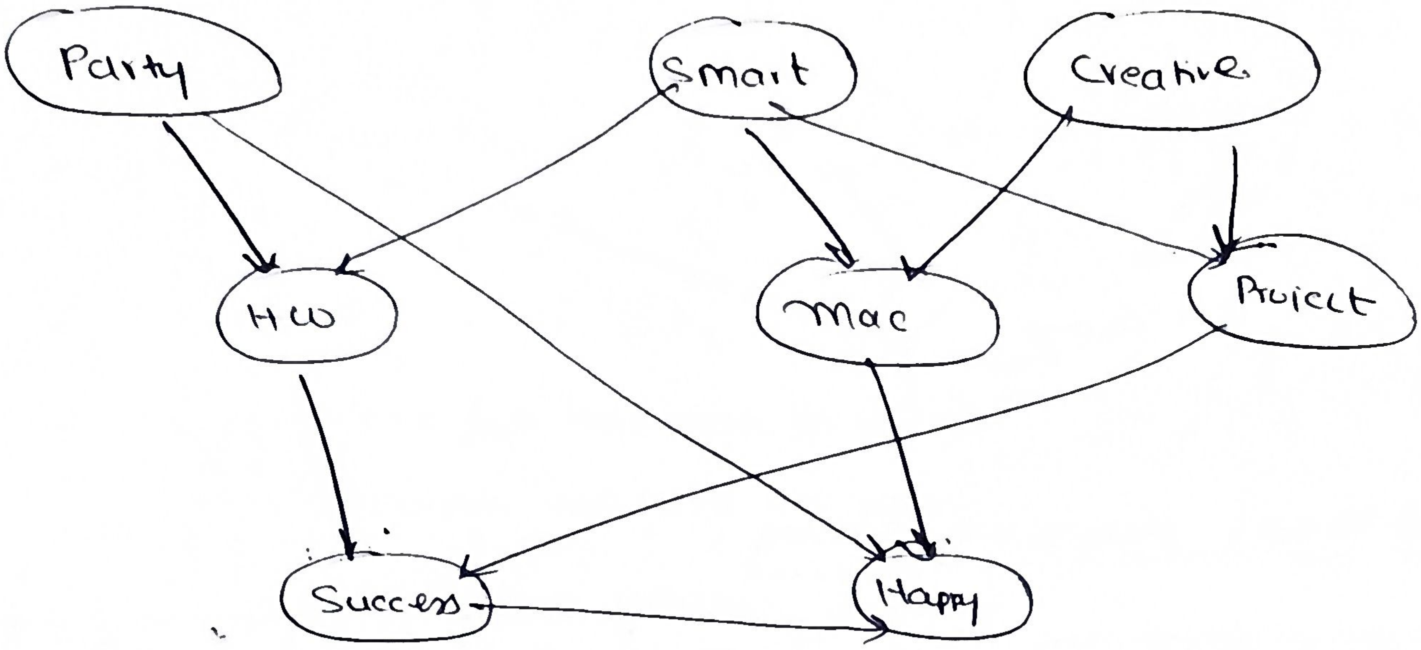
The joint probability is:

$$P(E, B, T, A) = P(E) \times P(T|E) \times P(B) \times P(A|E, T, B)$$

$$= 0.\cancel{8}^{4} \times 0.8 \times 0.7 \times 1.0$$

Example 2 - CAT-3Q8 - Construct a Bayesian network for the following set of conditional relationships using eight random variables

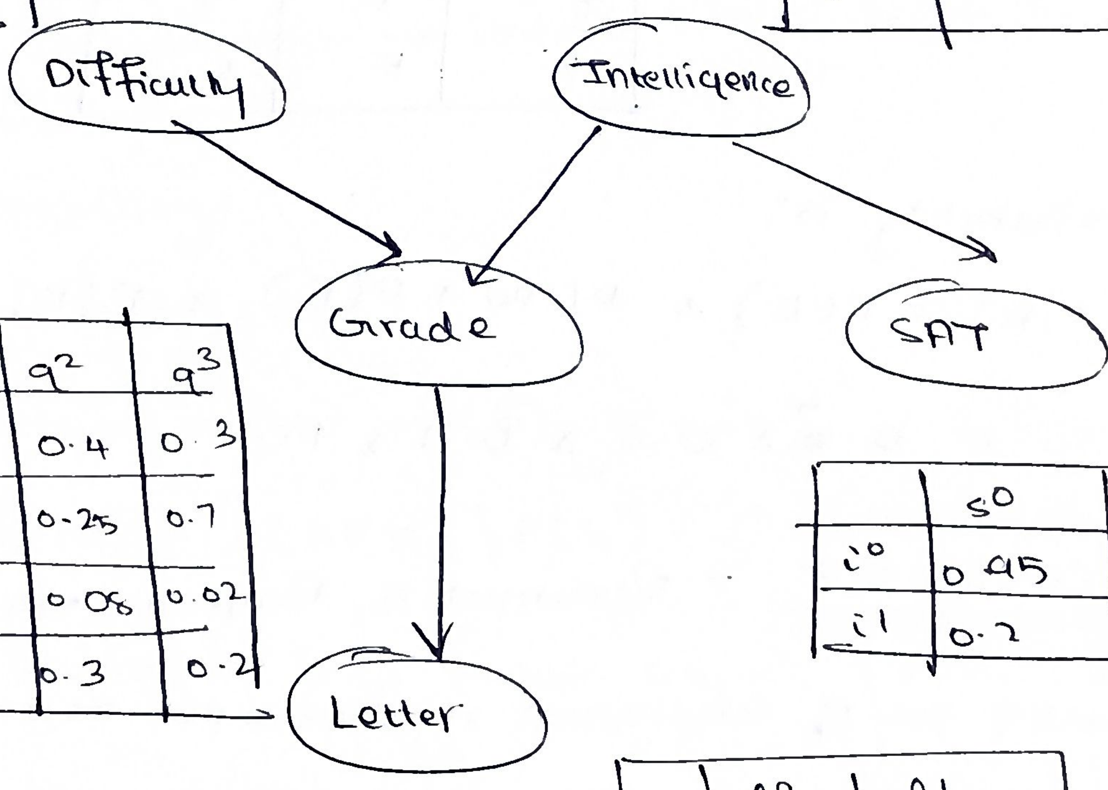
party, smart, creative, hw, mac, project, success, happy



Example 3: Find the joint probability value of the following Bayesian network.

d^0	d^1
0.6	0.4

i^0	i^1
0.7	0.3



	g^1	g^2	g^3
i^0, d^0	0.3	0.4	0.3
i^0, d^1	0.05	0.25	0.7
i^1, d^0	0.9	0.08	0.02
i^1, d^1	0.5	0.3	0.2

	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

	l^0	l^1
g^1	0.1	0.9
g^2	0.4	0.6
g^3	0.99	0.01

$$P(D) \times P(I) \times P(G \mid D, I) \times P(S \mid I) \times P(L \mid G)$$

Foundations of Artificial Intelligence

Unit 3

Knowledge Representation and Reasoning

Logical agents - knowledge-based agents - propositional logic - propositional theorem proving; First order logic : syntax and semantics for first order logic - using first order logic; inference in first order logic: propositional versus first order logic - unification and lifting - forward chaining - backward chaining - resolution.

* Propositional Logic

→ propositional logic is represented with the following:

(i) logical constants : true, false

(ii) propositional symbols : P, Q, S (atomic sentences)

(iii) wrapping paranthesis : (....)

(iv) connectives : $\wedge$ - and - conjunction

$\vee$ - or - disjunction

$\Rightarrow$ - implies - implication | conditional

$\Leftrightarrow$ - is equivalent - biconditional

$\neg$ - not - negation

(v) literal: atomic sentences or negated atomic sentences

→ Propositional logic is useful for showing key ideas and definitions, it uses a set of propositional symbols like P & Q.

→ The user defines the semantics of each propositional symbol, this would determine the interpretation of the sentence.

* Terminology:

A. Well-formed Formula

→ A well formed formula is defined as follows:

(i) each symbol is a sentence

(ii) if S is a sentence, then $\neg S$ is also a sentence

(iii) if S is a sentence, then (S) is a sentence

(iv) if S and T are sentences, then $(S \vee T)$, $(S \wedge T)$, $(S \rightarrow T)$ and $(S \leftrightarrow T)$ are all sentences

(v) A sentence results from a finite number of applications of the above rules.

B. Model (for a knowledge base) - is a possible world (assignment of truth values to propositional symbols), in which each sentence in the knowledge base is True.

C. Valid Sentence (Tautology) - a sentence that is True under all interpretations, no matter what the world is actually like or how the semantics are defined. eg. It is raining or it is not raining.

D. Inconsistent sentence (Contradiction) - A sentence that is false under all interpretations. The world is never like what it describes

eg. It is raining and it is not raining?

E. Satisfiability - There exists some interpretation in some world for which it is true.

* Inference Rules

→ Logical inference is used to create new sentences that logically follow from a given set of predicate calculus sentences (knowledge base).

Properties for Inference

A. Soundness - If $KB \vdash Q$ then $KB \models Q$

- If Q is derived from a set of sentences KB using a given set of rules of inference, then Q is entailed by KB.

- i.e the inference rule does not create any algorithms

B. Completeness - If $KB \models Q$, then $KB \vdash Q$

- If Q is entailed by a set of sentences KB, then Q can be derived from the KB using the rules of inference.

→ Some sound rules of inference are as follows:

RULE	PREMISE	CONCLUSION
Modus ponens	$A, A \rightarrow B$	B
And Introduction	A, B	$A \wedge B$
And Elimination	$A \wedge B$	A (or B)
Double Negation	$\neg\neg A$	A
Unit Resolution	$A \vee B, \neg B$	A
Resolution	$A \vee B, \neg B \vee C$	$A \vee C$

* Entailment and Derivation

Entailment - $KB \models Q$

- Q is entailed by the KB if and only if the conclusion is true in every logically possible world in which all the premises in KB are true.

Derivation - $KB \vdash Q$

- Q can be derived from the KB, if there is a proof consisting of valid inference steps starting from the premises in the KB and resulting in Q.

* Proofs

→ A proof is a sequence of sentences, where each sentence is either a premise or a sentence derived from earlier sentences in the proof by one of the rules of inference.

→ The last sentence is the theorem (the goal / query) that we want to prove.

* Horn Sentences

→ A horn sentence / clause has the form:

$$P1 \wedge P2 \wedge \ldots Pn \rightarrow Q$$

(or) $$\neg P1 \vee \neg P2 \vee \ldots \neg Pn \vee Q$$

→ To get a proof for Horn sentences, apply modus ponens repeatedly until nothing can be done

* Limitations of Propositional Logic

→ Propositional logic is considered a weak language because:

(i) it is hard to identify individuals

(ii) it can't directly talk about properties of individuals or relations between individuals

(iii) Generalizations, patterns, regularities can't easily be represented

(iv) impractical, only works for very small worlds, just suffices to illustrate the process of inference

→ First (OL) Order Logic is more expressive and helps overcome those limitations.

* Symbolic Representation of Inference Rules in Propositional Logic

$\frac{\alpha}{\beta}$ means that β can be derived from α by inference.

① Modus Ponens : $\frac{\alpha, \alpha \Rightarrow \beta}{\beta}$

② And Elimination : $\frac{\alpha_1 \wedge \alpha_2 \wedge \ldots \wedge \alpha_n}{\alpha_1}$

③ And Introduction : $\frac{\alpha_1, \alpha_2 \ldots \alpha_n}{\alpha_1 \wedge \alpha_2 \wedge \ldots \alpha_n}$

④ Or Introduction : $\frac{\alpha_1}{\alpha_1 \vee \alpha_2 \vee \alpha_3 \ldots \alpha_n}$

⑤ Double Negation Elimination : $\frac{\neg\neg\alpha}{\alpha}$

(6) unit resolution : $\dfrac{\alpha \vee \beta, \neg\beta}{\alpha}$

(7) resolution : $\dfrac{\alpha \vee \beta, \neg\beta \vee \gamma}{\alpha \vee \gamma}$

* Knowledge-based Agents

→ A knowledge-based agent has to perform tasks using logic representation. The tasks are:

(i) to know the current state of the world

(ii) how to infer the unseen properties of the world

(iii) new changes in the environment

(iv) goal of the agent

(v) how to perform actions based on circumstances

→ The component of a knowledge-based agent is its knowledge base or KB.

→ Each individual representation is called a sentence. The sentences are expressed in a language called a knowledge representation language.

→ There are 3 levels of knowledge-based agents:

(i) Knowledge level - an external level which describes the agent by saying what it knows as a formal language.

(ii) logical level - the level at which the knowledge is encoded into logical sentences

(iii) implementation level - the level that runs on the agent architecture

The Wumpus World Environment

Problem Statement - based on an agent who explores a cave consisting of rooms connected by passage ways.

→ Somewhere in the cave, there is a wumpus, a beast that eats anyone who enters its room.

→ Some rooms contain bottomless pits, that the agent can fall into. (The wumpus can't, since it is too big to fall in).

→ The only mitigating feature of the environment is the occasional heap of gold.

Given - (i) The wumpus world is a grid of squares surrounded by walls.

(ii) Each square contains agents and objects.

(iii) In the square containing the wumpus, and in the adjacent squares, the agent will perceive a stench.

(iv) In the square adjacent to a pit, the agent will perceive a breeze.

(v) In the square where the gold is, the agent will perceive a glitter.

(vi) When an agent walks into a wall, it will perceive a bump.

(vii) When the wumpus is killed, it will give a scream perceived from anywhere in the cave.

(vii) The percept is given in the following order:

[Stench, Breeze, Glitter, Bump, Scream]

Operators . (i) Move Forward

(ii) ~~Move~~ Turn left by 90°

(iii) Turn right by 90°

(iv) Grab - pick up an object

(v) Shoot - fire an arrow to kill the wumpus, can be used only once.

(vi) Climb - leave the cave, from the start square

Initial state : agent at (1,1)

Goal state - to find the goal, and bring it out of the cave as quickly as possible, without getting killed.

4	Stench		Breeze	PIT
3	Wumpus	Breeze Stench Gold	PIT	Breeze
2	Stench		Breeze	
1	Agent (start)	Breeze	PIT	Breeze
	1	2	3	4

* First Order / Predicate Logic

→ FOL models the world in terms of :

(i) objects - individual entities

(ii) properties - what distinguishes them from other objects

(iii) relations - that hold among sets of objects

(iv) functions - a subset of relations where there is only one value for any given input.

→ The user would provide:

(i) constants - Mary, 3, green

(ii) function symbols - map individuals to individuals

- father-of (Mary) = John

(iii) predicate symbols - greater (5, 3), green (Grass)

→ FOL provides

(i) variables

(ii) connectives: $\neg, \wedge, \vee, \rightarrow, \Leftrightarrow$

(iii) quantifier - universal, $(\forall x)$

- existential, $(\exists x)$

* Well-formed formula (WFF)

→ A well formed formula is a sentence that contains no free variables

→ All variables are bound by universal or existential quantifiers.

* Quantifiers

A. Universal Quantification

→ $(\forall x)P(x)$ means that P holds for all values of x in the domain associated with that variable.

→ They are used with implies. to form rules

eg. $(\forall x)\text{student}(x) \longrightarrow \text{smart}(x)$

All students are smart

B. Existential Quantification

→ $(\exists x) P(x)$ means that P holds for some value of x in the domain associated with that variable.

→ They are used with 'and' to specify a list of properties about an individual.

eg. $\exists x)$ student$(x) \wedge$ smart(x) means There is a student who is smart.

* Quantifier Scope

1. Switching the order of universal quantifiers does not change the meaning.

2. Switching the order of existential quantifiers also does not change the meaning.

3. Switching the order of universal & existential quantifiers does change the meaning:

(i) everyone likes someone: $(\forall x)(\exists y)$ likes (x, y)

(ii) someone is liked by everyone: $(\exists y)(\forall x)$ likes (x, y).

* Connecting Universal & Existential Quantifiers

→ They can be related using De Morgan's laws:

(i) $(\forall x) \neg P(x) \leftrightarrow \neg(\exists x) P(x)$

(ii) $\neg(\forall x)P(x) \leftrightarrow (\exists x)\neg P(x)$

(iii) $(\forall x)P(x) \leftrightarrow \neg(\exists x)\neg P(x)$

(iv) $(\exists x)P(x) \leftrightarrow \neg(\forall x)\neg(P(x))$

* Quantified Inference Rules

(i) universal instantiation: $\forall x\, P(x)$ $\quad \therefore P(A)$

(ii) universal generalization: $P(A) \wedge P(B) \ldots$ $\quad \forall x\, P(x)$

(iii) existential instantiation: $(\exists x)\, P(x)$ $\quad P(F)$ → skolem constant

(iv) existential generalization: $P(A)$ $\quad (\exists x)\, P(x)$

* Skolemization

→ In existential instantiation, i.e $(\exists x) P(x)$ to infer $P(F)$, F is the skolem constant

→ Skolemization is the process of replacing the variable x, with a brand new constant, that does not occur in any sentence in the knowledge base.

→ This is to ensure that other inferences are not accidentally drawn from using existing constants.

→ It is convenient to use this new constant to reason about the unknown object, rather than constantly manipulating the existential quantifier.

* Differences between propositional and predicate logic

PROPOSITIONAL LOGIC	PREDICATE LOGIC
(i) Each fact is represented by one symbol	(i) Representation of world in terms of objects and predicates on objects (i.e properties of objects or relations between objects)
(ii) Propositional symbols can be connected with boolean connectives, to give more complex meaning - like $\vee, \wedge, \Rightarrow, \Leftrightarrow, \neg$	(ii) Connectives and quantifiers are used to represent the meaning of the sentence. Connectives: $\wedge, \vee, \neg, \Rightarrow, \Leftrightarrow$ Quantifiers: $\forall, \exists$
(iii) Simple statements are implemented.	(iii) Complex statements are implemented
(iv) eg. PLATOMAN	(iv) eg. MAN(PLATO)

* Inference in FOL - Unification, Forward and Backward Chaining

Universal Instantiation

→ Every instantiation of a universally quantified sentence is entailed by it.

$$\frac{\forall v \quad \alpha}{\text{subst}(\{v/g\}, \alpha)}$$

eg. $(\forall x)$ king(x) $\wedge$ greedy(x) $\Rightarrow$ evil(x) gives

king(John) $\wedge$ greedy(John) $\Rightarrow$ Evil(John)

Existential Instantiation

For any sentence α, variable v, and constant k that does not appear in the KB.

skolem

$$\frac{\exists v \quad \alpha}{\text{subst}(\{v/k\}, \alpha)}$$

eg. $\exists x$ crown(x) $\wedge$ onhead$(x$, John$)$ gives

crown(k) $\wedge$ onhead$(k$, John$)$

Unification

Unification - The process of finding all legal substitutions that make logical expressions look identical

→ a recursive algorithm

→ For example - in the given sentence - we can get the inference immediately if there is a substitution θ such that King(x) and greedy(x) match King(John) and Greedy(y)

$\theta = \{x/\text{John}, y/\text{John}\}$ unification would mean unify$(\alpha, \beta) = \theta$ if $\alpha\theta = \beta\theta$

Examples

P	q	Q
knows(John, x)	Knows (John | Jane)	{x | Jane}
Knows (John, x)	Knows (y, OJ)	{x | OJ, y | John}
Knows (John, x)	Knows (y, mother(y))	{y | John, x | mother (John)}

* Algorithm for Unification

Function UNIFY (x, y, θ) returns a substitution to make x & y identical

inputs : x, a variable | constant | list | compound

y, a variable | constant | list | compound

θ, the substitution built up so far

if θ = failure, then return failure

else if x = y, return θ

else if VARIABLE? (x) return UNIFY-VAR (x, y, θ)

else if VARIABLE ? (y) 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

inputs : var - a variable

x, any expression

θ, substitution built up so far

if {var/val} ∈ θ, return UNIFY (val, x, θ)

else if {x/val} ∈ θ, return UNIFY (var, val, θ)

else if OCCUR-CHECK? (var, x) then return failure

else return {var/x} to θ

* Forward Chaining

→ Forward chaining is a form of reasoning which starts with atomic sentences in the knowledge base, and applies inference rules in the forward direction to extract more data until a goal is reached.

→ It is a down-up approach.

→ It starts from known facts, triggers all rules whose premises are satisfied, and add their conclusion to the known facts.

Algorithm

function FOL-FC-ASK (KB, α) returns a substitution or false

repeat until new is empty

new ← { }

for each sentence r in KB do:

$(p_1 \wedge p_2 \ldots \wedge p_n \} \Rightarrow q)$ ← standardize-apart(r) for each θ such that $(p_1 \wedge \ldots \wedge p_n)\theta = (p_1' \wedge \ldots \wedge p_n')\theta$

for some $p_1', \ldots, p_n'$ in KB

$q' \leftarrow$ SUBST (θ, q)

If q' is not a renaming of a sentence already in KB

or new then do

add q' to new

$\phi \leftarrow$ UNIFY (q', α)

if ϕ is not fail then return ϕ

add new to KB

return false

* Backward Chaining?

→ A backward chaining algorithm starts with the goal and works backwards, chaining through rules to find known facts that support the goal.

→ It is a top-down approach

→ It is a goal-driven approach, as a list of goals decides whether a rule is selected and used.

Algorithm

function FOL-BC-ASK (KB, goals, θ) returns a set of substitutions

inputs: KB, a knowledge base

goals, a list of conjuncts forming a query

θ, the current substitution, initially empty if goals is empty then return {θ}

q' ← SUBST(θ, FIRST(goals))

for each r in KB where STANDARDIZE-APART(r) =

(p1 ∧ p2 ... pn) ⇒ q)

and θ' ← UNIFY(q, q') succeeds

ans ← FOL-BC-ASK(KB, [p1, ... pn], REST(goals)

COMPOSE(θ', θ)) ∪ ans

return ans

Properties of backward chaining?

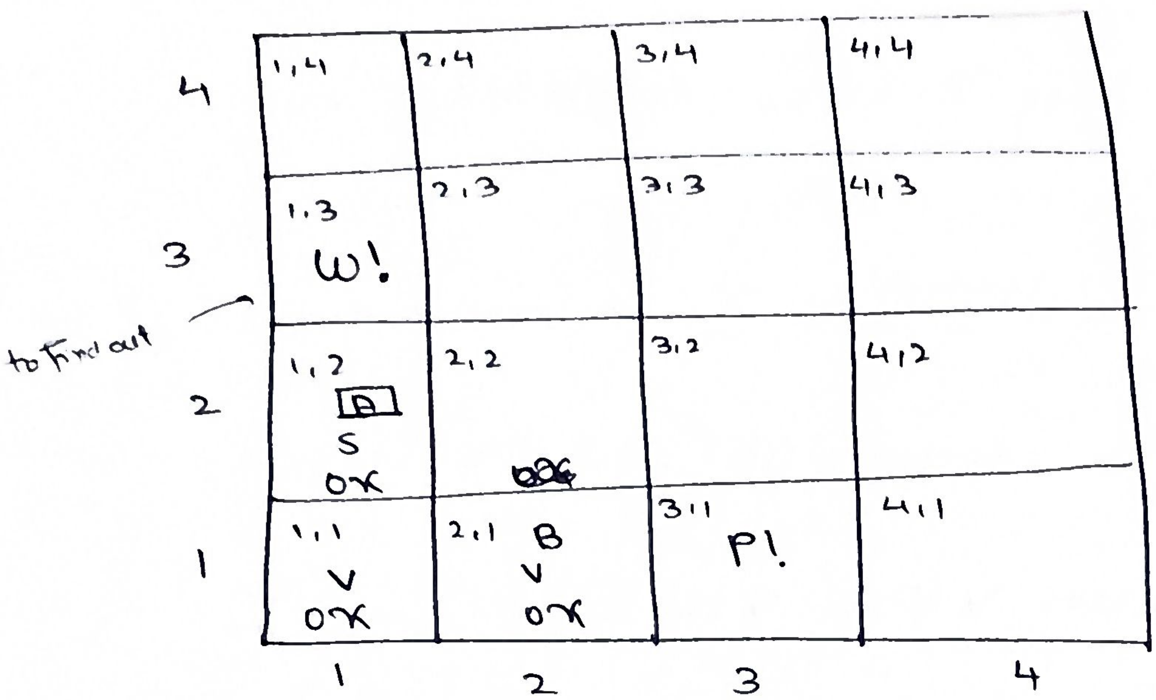
→ It uses DFS - space is linear

→ Incomplete due to infinite loops - (check current goal w/ every subgoal on the stack)

→ Inefficient due to repeated subgoals (use caching of previous results)

Unit 3- Problems

(Q1) Given the following wumpus world environment, deduce the presence of the wumpus, using inference rules.



Ans Knowledge base

(i) S_{12} - stench in (1,2)

(ii) $\neg B_{1,2}$ - no breeze in (1,2) } (1,2)

(iii) $S_{12}, \neg B_{1,2}$

(iv) $\neg S_{1,1}$

(v) $\neg B_{1,1}$ } (1,1)

(vi) $\neg S_{11}, \neg B_{11}$

(vii) B_{21}

(viii) $\neg S_{21}$ } (2,1)

(ix) $\neg S_{21}, B_{21}$

Rules from knowledge base

R_1: $\neg S_{11} \Rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21}$

R_2: $\neg S_{21} \Rightarrow \neg W_{11} \wedge \neg W_{22} \wedge \neg W_{21} \wedge \neg W_{31}$

R_3: $S_{12} \Rightarrow W_{13} \vee W_{11} \vee W_{22} \vee W_{12}$

Inference Rules to find the wumpus

① Apply modus ponens to sentence (ii) and R_3

$$\frac{S_{12}, \quad S_{12} \Rightarrow W_{13} \vee W_{11} \vee W_{22} \vee W_{12}}{W_{13} \vee W_{11} \vee W_{22} \vee W_{12}} \quad - (A)$$

② Apply modus ponens to sentence (iv) and R_1

$$\frac{\neg S_{1}, \quad \neg S_{11} \Rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21}}{\neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21}} \quad - (B)$$

③ Apply modus ponens to sentence (viii) and R_2

$$\frac{\neg S_{21} \quad \neg S_{21} \Rightarrow \neg W_{11} \wedge \neg W_{22} \wedge \neg W_{21} \wedge \neg W_{31}}{\neg W_{11} \wedge \neg W_{22} \wedge \neg W_{21} \wedge \neg W_{31}} \quad - (C)$$

④ Apply AND- elimination to (B)

$\neg W_{11}$, $\neg W_{12}$, $\neg W_{21}$ — (D)

⑤ Apply AND elimination to (C)

$\neg W_{11}$, $\neg W_{22}$, $\neg W_{21}$, $\neg W_{31}$ - (E)

⑥ Apply unit resolution to (A) and $\neg W_{11}$ (from D)

unit resolution:

$$\frac{\alpha \vee \beta, \neg\beta}{\alpha}$$

$$\frac{W_{13} \vee W_{22} \vee W_{12} \vee W_{11}, \neg W_{11}}{W_{13} \vee W_{22} \vee W_{12}} \quad \text{-(F)}$$

⑦ Apply unit resolution to (F) and $\neg W_{12}$ (from D)

$$\frac{W_{13} \vee W_{22} \vee W_{12}, \neg W_{12}}{W_{13} \vee W_{22}} \quad \text{-(G)}$$

⑧ Apply unit resolution to (G) and $\neg W_{22}$ from (E)

$$\frac{W_{13} \vee W_{22}, \neg W_{22}}{W_{13}}$$

$\Rightarrow$ $\boxed{\text{The wumpus is in } W_{13}}$

(Q2) Convert the following English sentences to FOL.

(i) Every gardener likes the sun

Ans. $(\forall x)$ gardener$(x) \longrightarrow$ likes(x, sun)

(ii) You can fool some of the people all of the time.

Ans $(\exists x)(\forall t)$ person$(x) \wedge$ time$(t) \rightarrow$ canfool(x, t)

(iii) You can fool all the people some of the time.

$(\forall x)(\exists t)(person(x) \rightarrow can\text{-}fool(x,t) \wedge time(t))$

(iv) All purple mushrooms are poisonous.

$(\forall x)(mushroom(x) \wedge purple(x)) \rightarrow poisonous(x))$

(v) No purple mushroom is poisonous

$(\forall x)(mushroom(x) \wedge purple(x)) \rightarrow \neg poisonous(x))$

(vi) Clinton is not tall

$\neg tall(Clinton)$

★★★

(vii) X is above Y iff X is on directly on top of Y or there is a pile of one or more other object directly on top of one another starting with X and ending with Y

$(\forall x)(\forall y)\ above(x,y) \leftrightarrow (on(x,y) \cancel{\wedge\ above(z,y)})$

$\vee\ \exists z(on(x,z) \wedge above(z,y)))$

(viii) Everyone likes McDonald's.

$(\forall x)\ likes(x, McDonald's)$

(ix) Someone likes McDonald's

$(\exists x)\ likes(x, McDonald's)$

(x) All children like McDonalds

$(\forall x)\ children(x) \rightarrow likes(x, McDonald's)$

(xi) Everyone likes McDonald's unless they are allergic to it

$\forall(x)$ likes $(x, \text{McDonald's}) \vee$ allergic $(x, \text{McDonald's})$

(xii) • Everyone likes some kind of food.

$\forall x \; \exists y$ food$(y) \wedge$ likes (x, y)

• There is a kind of food that everyone likes.

$\exists x \; \forall y$ food$(x) \wedge$ likes(y, x)

• Someone likes all kinds of food

$\exists(x) \; \forall(y)$ food$(y) \wedge$ likes(x, y)

• Every food has someone who likes it.

$\forall(x) \; \exists(y)$ food$(x) \wedge$ likes(y, x)

(Q3) Express the following sentences using both the universal and existential quantifier.

1. Not everyone likes McDonald's.

$\neg((\forall x)$, likes $(x, \text{McDonald's}))$

$(\exists x) \neg$ likes $(x, \text{McDonalds})$

2. No one likes McDonald's

$\neg((\exists x)$, likes$(\text{McDonalds}))$

$(\forall x) \neg$ likes (McDonalds)

(Q4) Apply forward chaining ~~for~~ the to derive the following conclusion

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

Prove that Col. West is a criminal.

Ans

1. It is a crime for an American to sell weapons to hostile nations

American(x) ∧ Weapon(y) ∧ Sells(x, y, z) ∧ Hostile(z)

⇒ criminal(x)

2. Nono has some missiles

∃x Owns(Nono, x) ∧ missile(x)

⇒ owns(Nono, M1) and Missile(M1)

3. All of its missiles were sold to it by Col. West

∀x Missile(x) ∧ owns(Nono, x) ⇒ sells(ColWest, x, Nono)

4. Missiles are weapons

missile(x) ⇒ weapon(x)

5. An enemy of America counts as hostile

enemy(x, America) ⇒ hostile(x)

6. Col. West who is an American

American (Col west)

7. The country , Nono, is an enemy of America

Enemy (Nono, America)

Forward chaining

Level 1

American (West) | Missile(M1) | Owns (Nono, M1) | Enemy (Nono, America)

Rule 3 is satisfied by {x | M1} , add the result

Level 2

sells (West, M1, Nono)

American(West) | Missile (M1) | Owns (Nono, M1) | Enemy(Nono, America)

Rule 5 is satisfied with the substitution {x| Nono}

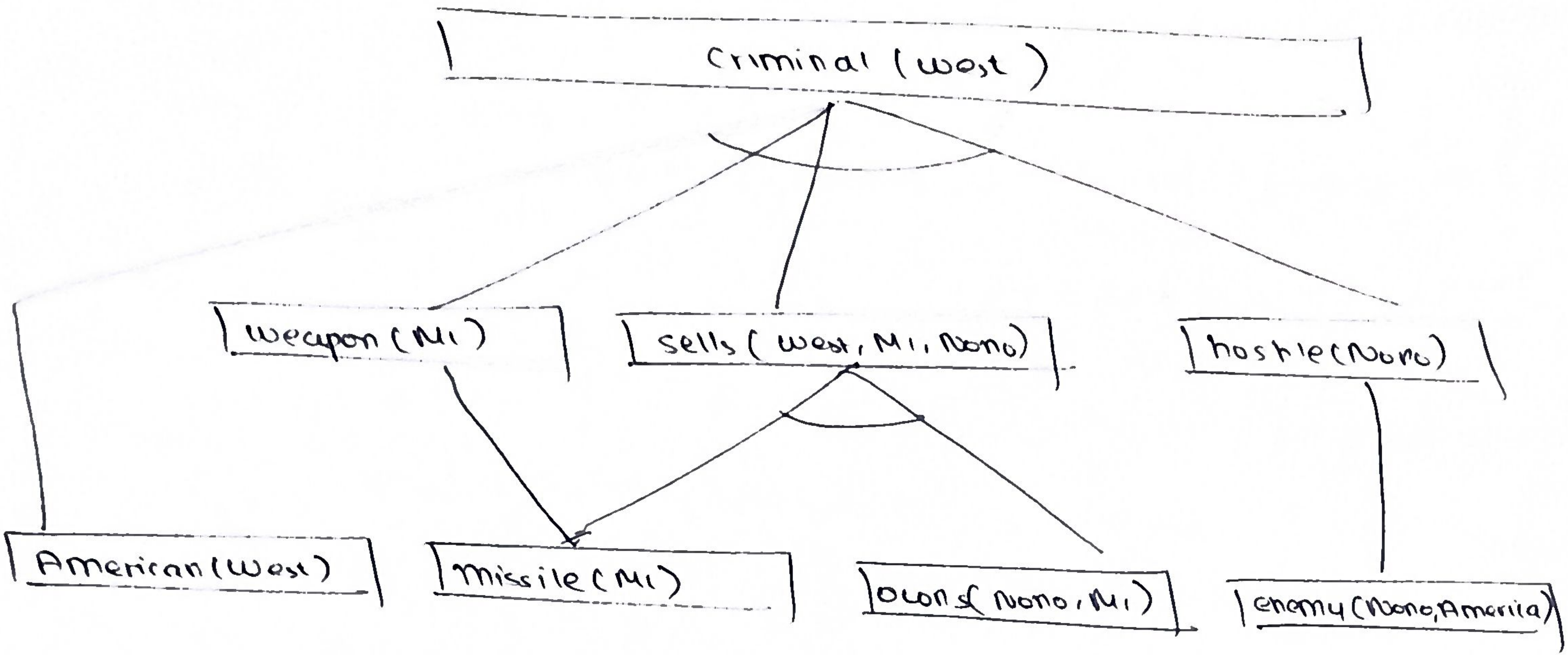
Rule 4 is satisfied with {x | M1}

Level 3

weapon (M1) | sells(West, M1, Nono) | hostile (Nono)

American(West) | Missile (M1) | owns(Nono, M1) | Enemy(Nono, America)

Level 4 - Rule 1 satisfies all the conditions



Backward chaining sub with variables in the same tree.

(Q5) Given the following rules and facts, prove the following using resolution.

Rules cold and precipitation → snow

January → cold

Clouds → precipitation

Facts January, clouds

Prove snow

Ans Convert to CNF

~~Read~~ (1) cold ∧ precipitation → snow

¬(cold ∧ precipitation) ∨ snow

= $\boxed{\neg \text{cold} \vee \neg \text{precipitation} \vee \text{snow}}$

(ii) January → cold

$\boxed{\neg \text{January} \vee \text{cold}}$ ✓

(iii) clouds ~~cold~~ → precipitation

$\boxed{\neg \text{clouds} \vee \text{precipitation}}$ ✓

Facts

~~snow~~ clouds

January

Result

snow =

⇓

¬snow

¬snow

snow ∨ ¬cold ∨ ¬precipitation

¬cold ∨ ¬precipitation

cold ∨ ¬January.

¬January ∨ ¬precipitation

¬ clouds ∨ precipitation

¬ January ∨ ¬ clouds

January

¬clouds

clouds

{ }

Q6 Convert to FOPL, then to CNF, and prove that Fido will die

1. All dogs are animals
2. Fido is a dog
3. All animals will die.

	CNF
1. $(\forall x)\ dog(x) \rightarrow animal(x)$	$\neg dog(x) \vee animal(x)$
2. $dog(Fido)$	$dog(Fido)$
3. $(\forall y)\ animal(y) \rightarrow die(y)$	$\neg animal(y) \vee die(y)$

To prove : Fido will die

$die(Fido)$

$\neg dog(x) \vee animal(x)$ $\qquad$ $\neg animal(y) \vee die(y)$

$\{x/y\}$

$\neg dog(y) \vee die(y)$ $\qquad$ $dog(Fido)$

$\{y/Fido\}$

$die(Fido)$ $\qquad$ $\neg dog(Fido)$

$\{\ \}$

$\therefore$ Fido will die.

(Q7) Using resolution, prove that John is happy?

1. Anyone passing his history exams and winning the lottery is happy

2. Anyone who studies or is lucky can pass all his exams

3. John did not study, but he is lucky

4. Anyone who is lucky wins the lottery

1. $(\forall x)$ pass(x, history) $\wedge$ win(x, lottery) $\rightarrow$ happy(x)

2. $(\forall x)(\forall y)$ studies(x) $\vee$ lucky(x) $\rightarrow$ pass(x, y)

3. $\neg$ studies(John) $\wedge$ lucky(John)

4. $(\forall x)$ lucky(x) $\rightarrow$ win(x, lottery)

in CNF

1. $\neg$(pass(x, history) $\wedge$ win(x, lottery)) $\vee$ happy(x)

(($\neg$pass(x, history) $\vee$ $\neg$win(x, lottery)) $\vee$ happy(x))

2. ($\neg$studies(x) $\wedge$ $\neg$lucky(x)) $\vee$ pass(x, y)

$\neg$studies(y) $\vee$ pass(y, z)

$\neg$ studies(w) $\vee$ pass(w, v)

3. $\neg$studies(John)

lucky(John)

4. $\neg$lucky(u) $\vee$ win(u, lottery)

$\neg$happy(John)

(8) Prove using resolution (CMU PPT)

1. Everyone who loves animals is loved by someone

2. Anyone who kills an animal is loved by no one

3. Jack loves all animals

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

Did curiosity kill the cat?

(1) $\Rightarrow (\forall x)[(\forall y)\ animal(y) \Rightarrow loves(x,y)] \Rightarrow \exists y\ loves(y,x)$

(2) $\forall x)[\exists z)\ animal(z) \wedge kills(x,z)] \Rightarrow \forall y\ \neg loves(y,x)$

(9) Prove w/ resolution

1. If it rains, Joe brings his umbrella

2. If Joe has an umbrella, he doesn't get wet

3. If it doesn't rain, Joe doesn't get wet

Conclusion: Joe doesn't get wet

It rains = p

Joe brings umbrella = q

gets wet = w

conc $\neg w$

(1). $p \rightarrow q$ $\neg p \vee q$

(2) $q \rightarrow \neg w$ $\neg q \vee \neg w$

(3) $\neg p \rightarrow \neg w$ $p \vee \neg w$

w $p \vee \neg w$

p $\neg p \vee q$

q $\neg q \vee \neg w$

$\boxed{\neg w}$, w $\square$

(10) Save the following

S1: Either you do HW, or you will flunk

S2: If you do not do your HW, then you will flunk

S1, you do your hw $= \sim p$

you will flunk $= q$

$\boxed{\sim p \vee q}$

S2: If you don't do HW $= p$

the you will flunk $= q$

$\boxed{p \rightarrow q}$

(11) Consider the following premises

(i) It is not sunny and it is colder than yday

(ii) We will go swimming only if it is sunny

(iii) If we don't go swimming, then we will take a canoe trip

(iv) If we take a canoe trip, then we will be home by sunset.

Conc

we will be home by sunset

Convert to propositional & derive the given conclusion

It is sunny: p

It is colder than yday: q

we will go swimming: s

we will take a canoe trip: c

we will be home by sunset: h

1. $\neg p \wedge q$
2. ~~sssp~~ $p \rightarrow s$
3. $\neg s \rightarrow c$
4. $c \rightarrow h$

conc h

proof $\neg p \wedge q = \neg p, q$

$\neg p, p \rightarrow s \Rightarrow \neg s$

$\neg s, \neg s \rightarrow c = c$

$c, c \rightarrow h = h$

(*) P.T Marcus hates Caesar using resolution

(1) Marcus was a man

(2) Marcus was a Pompeian

(3) All pompeians were Romans

(4) Caesar was a ruler

(5) All romans were either loyal to caesar or hated him

(6) Everyone is loyal to someone

(7) People only try to assassinate rulers they are not loyal to

(8) Marcus tried to assassinate Caesar

(9) All men are people

Ans 1. man(Marcus)

2. Pompeian (Marcus)

3. $(\forall x)$ Pompeian$(x) \rightarrow$ Roman(x)

4. Ruler (Caersar)

5. $(\forall x)$ Roman$(x) \Rightarrow$ loyalto$(x,$ Caersar$) \vee$ hated$(x,$ Caesar$)$

6. $(\forall x)(\exists y)$ loyalto(x,y)

7. $(\forall x)(\forall y)$ people$(x) \wedge$ tryto assassinate$(x,y) \wedge$ ruler$(y) \Rightarrow \neg$loyalto(x,y)

8. tryto assasinate (Marcus, Caesar)

9. $(\forall x)$man$(x) \Rightarrow$ people(x)

¬~~Hated (Marcus, Caesar~~)   Roman(x)

Conversion to CNF

1. man(Marcus)
2. Pompeion(Marcus)
3. ¬Pompeian(x) ∨ Roman(x)
4. Ruler(Caesar)
5. ¬Roman(x) ∨ loyalto(x, caesar) ∨ hated(x, caesar)
6. loyalto(x, y)
7. ¬people(x) ∨ ¬trytoassassinate(x, y) ∨ ¬ruler(y) ∨ ¬loyalto(x, y)
8. trytoassassinate(Marcus, Caesar)
9. ¬man(x) ∨ people(x)

¬hated (Marcus, Caesar)   ¬roman(x) ∨ loyalto(x, caesar) ∨ hated(x, caesar)

¬roman(x) ∨ loyalto(x, caesar)   Roman(x) ∨ ¬pompeian(x)

loyalto(x, caesar) ∨ ¬pompeian(x)   pompeian(Marcus)

loyalto(Marcus, Caesar) with 7

w/ 8   ¬people(Marcus) ∨ ¬trytoassassinate(Marcus, Caesar) ∨ ¬ruler(Caesar)

¬people(Marcus) ∨ ¬ruler(Caesar)   w/ 4

¬people(Marcus)   w/ 9

¬man(x)   w/ w 1

{ } ⇒ hence proved

Q. Translate the following into FOL & prove that West is a criminal using resolution.

1. It is a crime for an American to sell weapons to hostile nations

2. Nono has some missiles

3. All of Nono's missiles were sold to it by Colonel West

4. Missiles are weapons

5. Enemies of America are hostile nations

6. Colonel west is an American

7. Nono is an enemy of American

① $(\forall x)(\forall y)(\forall z)$ American(x) $\wedge$ weapon(y) $\wedge$ hostilenation(z) $\wedge$

sell(x, y, z) $\Rightarrow$ criminal(x)

② owns(Nono, x) $\wedge$ missiles(x)

③ $(\forall x)$ missile(x) $\wedge$ owns(Nono, x) $\Rightarrow$ sells(West, x, Nono)

④ ~~weapo~~ $(\forall x)$ missile(x) $\Rightarrow$ weapon(x)

⑥ American(West)

⑤ $(\forall x)$ Enemy(x, America) $\Rightarrow$ hostilenation(x)

Enemy(nono, America)

~American (x) V ~weapon(u) V hostilenation(z) V ~sells(x,u,z) V criminal(x)

~ criminal(west)

~American (West) V ~ weapon (u) V ~ hostilenation(z) V ~sells (West,u,z)

7missile(x) V 7owns(Nono,x) V sells (West, x, Nono)

~American (West) V ~ weapon(u) V ~hostilenation(z) V ~7missile(x) V ~ owns(Nono,x)

American West)

~weapons(u) V ~hostilenation(x) V ~7missile(x) V ~owns (Nono,x)

⋮