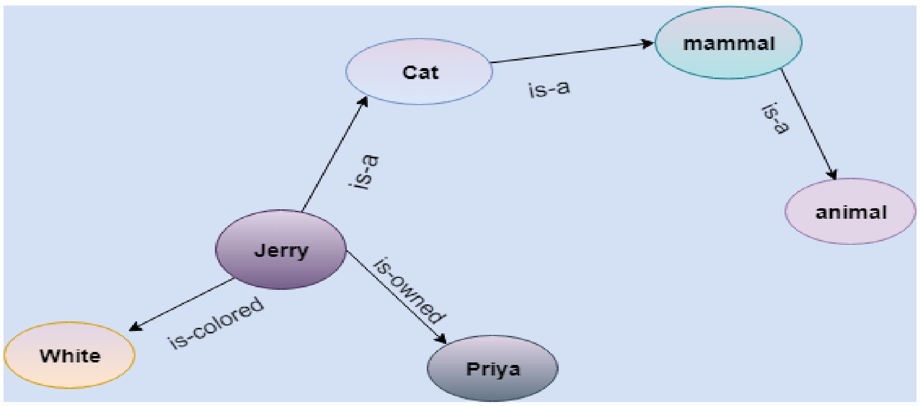
Sample Problems and Solutions

Problem: Following are some statements which we need to represent in the form of nodes and arcs.

Statements:

a) Jerry is a cat.
b) Jerry is a mammal
c) Jerry is owned by Priya.
d) Jerry is brown colored.
e) All Mammals are animal.

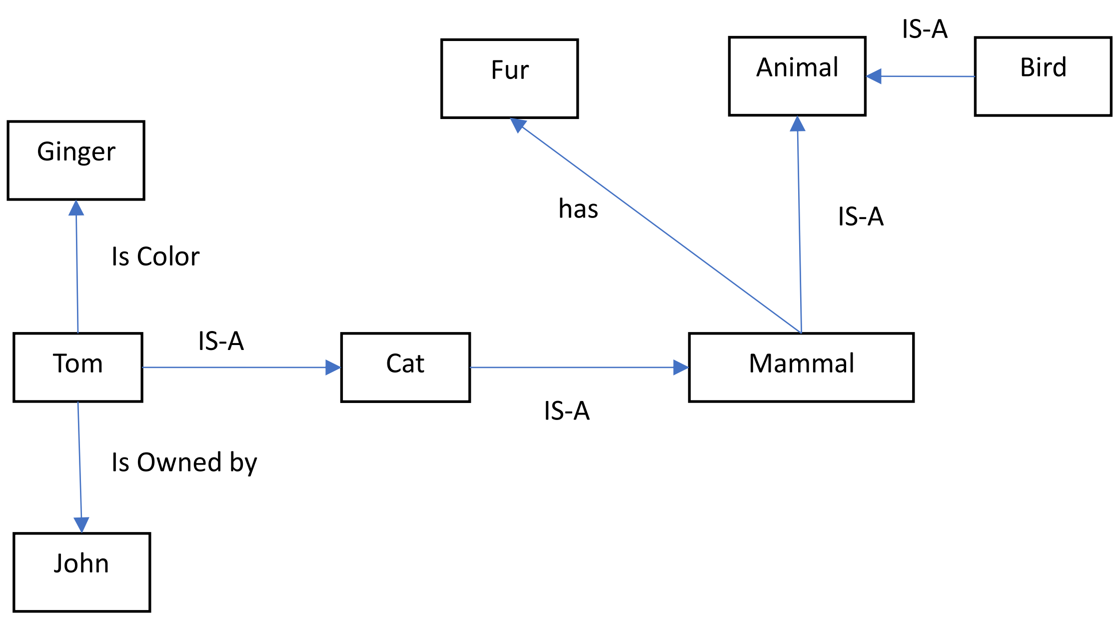
Solution:



Problem 1: Draw the semantic network that represents the data given below:

Mammals have fur. All mammals are animals. A bird is an animal. A cat is a mammal. Tom is a cat. Tom is owned by John. Tom is ginger in color.

Solution:



Problem: Find whether the meaning of the statement is true or false. “If the earth moves round the sun or the sun moves round the earth, then Copernicus might be a mathematician but was not an astronomer.”

Solution:

P = "The Earth moves round the Sun." = T

Q = "The Sun moves round the Earth." = T

M = "Copernicus is a mathematician." = T

A = "Copernicus is an astronomer." = T

$(T \lor T) \rightarrow (T \land \neg T)$

$= T \rightarrow (T \land F)$

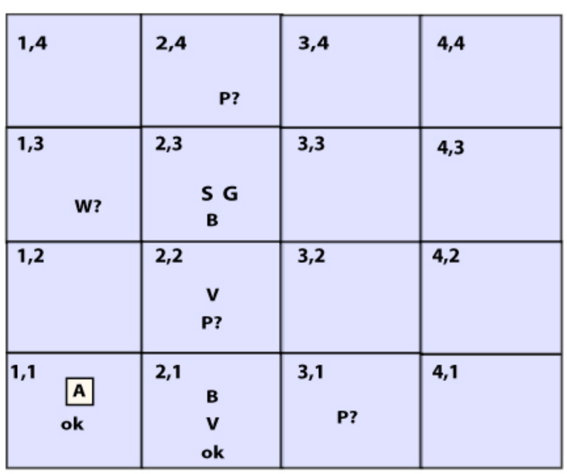
$= T \rightarrow F$

$= F$

So the meaning of the statement is False

Problem- Exploring Wumpus World:

We create a knowledge base for the wumpus world, and will derive some proves for the Wumpus-world using propositional logic. The agent starts visiting from first square [1, 1], and we already know that this room is safe for the agent. To build a knowledge base for wumpus world, we will use some rules and atomic propositions. We need symbol [i, j] for each location in the wumpus world, where i is for the location of rows, and j for column location.



Atomic proposition variable for Wumpus world:

- Let $\mathbf{P}_{i,j}$ be true if there is a Pit in the room [i, j].

- Let $\mathbf{B_{i,j}}$ be true if agent perceives breeze in [i, j], (dead or alive).
- Let $\mathbf{W_{i,j}}$ be true if there is wumpus in the square[i, j].
- Let $\mathbf{S_{i,j}}$ be true if agent perceives stench in the square [i, j].
- Let $\mathbf{V_{i,j}}$ be true if that square[i, j] is visited.
- Let $\mathbf{G_{i,j}}$ be true if there is gold (and glitter) in the square [i, j].
- Let $\mathbf{OK_{i,j}}$ be true if the room is safe.

Some Propositional Rules for the wumpus world:

(R1) $\neg S_{11} \Rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21}$

(R2) $\neg S_{21} \Rightarrow \neg W_{11} \wedge \neg W_{21} \wedge \neg W_{22} \wedge \neg W_{31}$

(R3) $\neg S_{12} \Rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{22} \wedge \neg W_{13}$

(R4) $S_{12} \Rightarrow W_{13} \vee W_{12} \vee W_{22} \vee W_{11}$

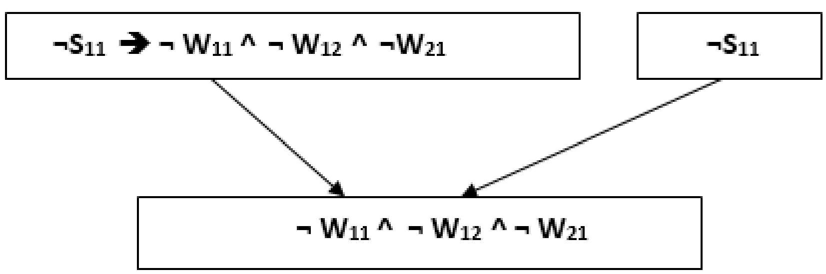
Prove that Wumpus is in the room (1, 3)

Solution:

We can prove that wumpus is in the room (1, 3) using propositional rules which we have derived for the wumpus world and using inference rule.

Apply Modus Ponens with ¬S11 and R1:

We will firstly apply MP rule with R1 which is $\neg S_{11} \rightarrow \neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21}$, and $\mathbf{\neg S_{11}}$ which will give the output $\neg W_{11} \wedge W_{12} \wedge W_{12}$.

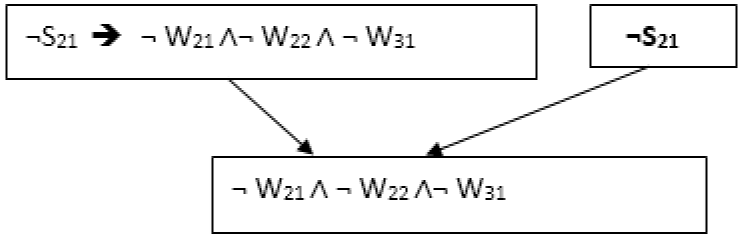


Apply And-Elimination Rule:

After applying And-elimination rule to $\neg W_{11} \wedge \neg W_{12} \wedge \neg W_{21}$, we will get three statements: $\mathbf{\neg W_{11}}$**,** $\mathbf{\neg W_{12}}$**, and** $\mathbf{\neg W_{21}}$.

Apply Modus Ponens to $\mathbf{\neg S_{21}}$**, and R2:**

Now we will apply Modus Ponens to $\neg S_{21}$ and R2 which is $\neg S_{21} \rightarrow \neg W_{21} \wedge \neg W_{22} \wedge \neg W_{31}$, which will give the Output as $\mathbf{\neg W_{21} \wedge \neg W_{22} \wedge \neg W_{31}}$

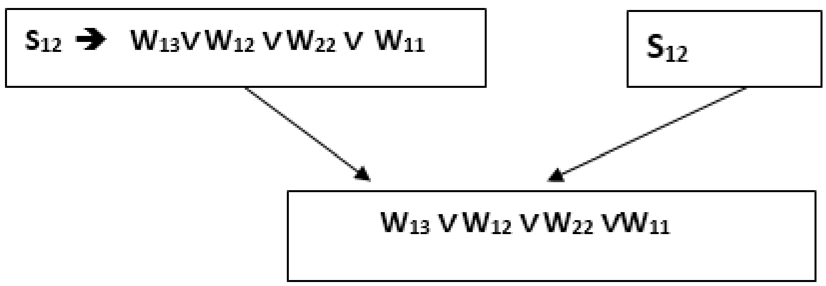


Apply And -Elimination rule:

Now again apply And-elimination rule to $\neg W_{21} \wedge \neg W_{22} \wedge \neg W_{31}$, we will get three statements: $\neg W_{21}$, $\neg W_{22}$, **and** $\neg W_{31}$.

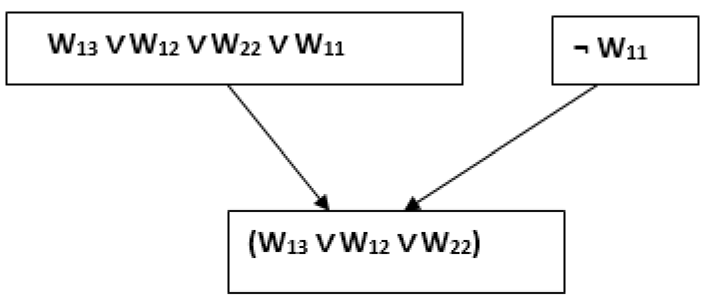
Apply MP to S_{12} and R4:

Apply Modus Ponens to S_{12} and R_4 which is $S_{12} \rightarrow W_{13} \vee W_{12} \vee W_{22} \vee W_{11}$, we will get the output as $W_{13} \vee W_{12} \vee W_{22} \vee W_{11}$.



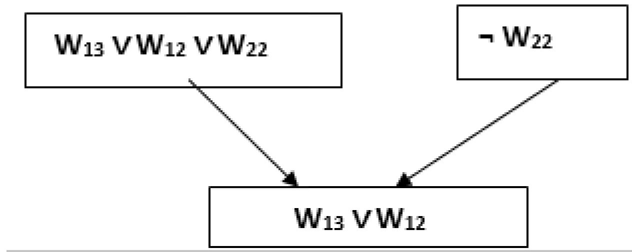
Apply Unit resolution on $W_{13} \vee W_{12} \vee W_{22} \vee W_{11}$ and $\neg W_{11}$:

After applying Unit resolution formula on $W_{13} \vee W_{12} \vee W_{22} \vee W_{11}$ and $\neg W_{11}$ we will get $W_{13} \vee W_{12} \vee W_{22}$.



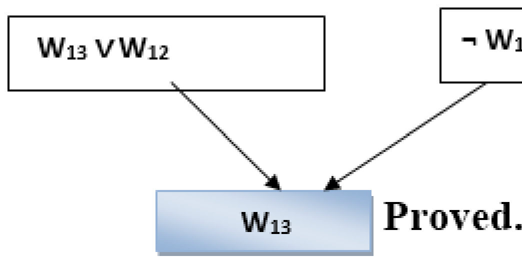
Apply Unit resolution on $W_{13} \vee W_{12} \vee W_{22}$ and $\neg W_{22}$:

After applying Unit resolution on $W_{13} \vee W_{12} \vee W_{22}$, and $\neg W_{22}$, we will get $W_{13} \vee W_{12}$ as output.



Apply Unit Resolution on $W_{13} \vee W_{12}$ and $\neg W_{12}$:

After Applying Unit resolution on **$W_{13} \vee W_{12}$ and $\neg W_{12}$**, we will get **W_{13}** as an output, hence it is proved that the Wumpus is in the room [1, 3].



Problem 2 - Wumpus World Problem

We can construct a knowledge base for the wumpus world. We focus first on the immutable aspects of the wumpus world, leaving the mutable aspects for a later section. For now, we need the following symbols for each [x, y] location: (Section 7.4. Propositional Logic: A Very Simple Logic Page 247)

- $P_{x,y}$ is true if there is a pit in [x, y].
- $W_{x,y}$ is true if there is a wumpus in [x, y], dead or alive.
- $B_{x,y}$ is true if the agent perceives a breeze in [x, y].
- $S_{x,y}$ is true if the agent perceives a stench in [x, y].

The sentences we write will suffice to derive $\neg P_{1,2}$ (there is no pit in [1,2]), We label each sentence R_i so that we can refer to them:

- There is no pit in [1,1]:
  - $R_1 : \neg P_{1,1}$.
- A square is breezy if and only if there is a pit in a neighboring square. This has to be stated for each square; for now, we include just the relevant squares:
  - $R_2 : B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$.
  - $R_3 : B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$.
- The preceding sentences are true in all wumpus worlds. Now we include the breeze percepts for the first two squares visited in the specific world the agent is in, leading up to the situation in Figure 7.3(b).
  - $R_4 : \neg B_{1,1}$.
  - $R_5 : B_{2,1}$.

Prove that no Pit at [1,2] $\neg P_{1,2}$.

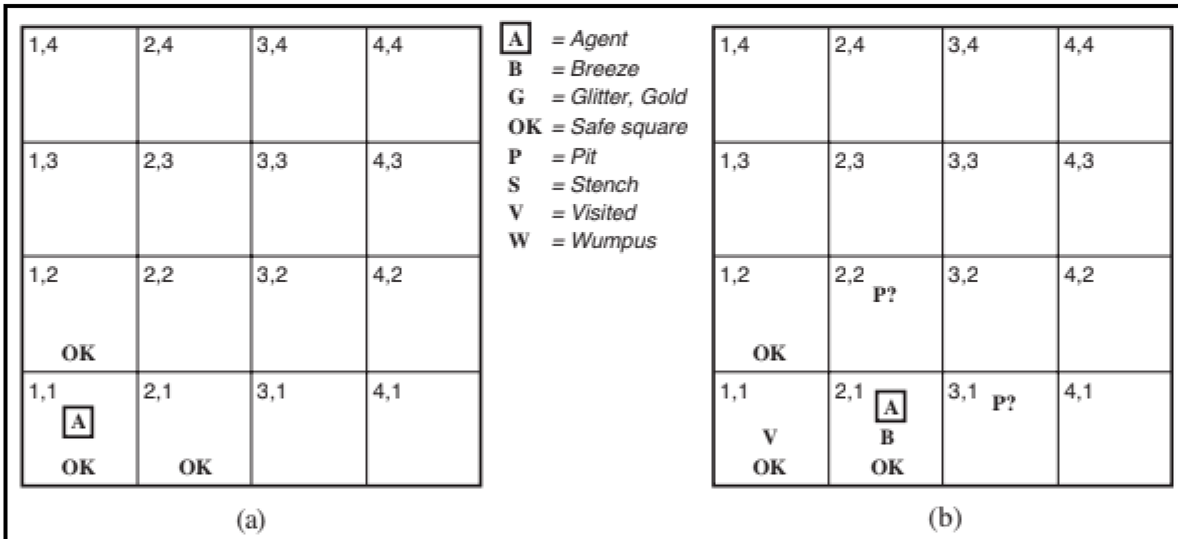


Figure 7.3 The first step taken by the agent in the wumpus world. (a) The initial situation, after percept [*None*, *None*, *None*, *None*, *None*]. (b) After one move, with percept [*None*, *Breeze*, *None*, *None*, *None*].

Solution:

We start with the knowledge base containing R1 through R5 and show how to prove $\neg P_{1,2}$, that is, there is no pit in [1,2].

First, we apply biconditional elimination to R2 to obtain

R2 : $B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$

Convert to, $B_{1,1} \Rightarrow (P_{1,2} \lor P_{2,1})) \land ((P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1}$

Then we apply And-Elimination to obtain

$(P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1}$

Logical equivalence for contrapositives gives (from propositional rules)

$\neg B_{1,1} \Rightarrow \neg(P_{1,2} \lor P_{2,1})$

Now we can apply Modus Ponens and the percept R4 (i.e., $\neg B_{1,1}$), to obtain

$\neg(P_{1,2} \lor P_{2,1})$

Finally, we apply De Morgan's rule, giving the conclusion

$\neg P_{1,2} \land \neg P_{2,1}$.

That is, neither [1,2] nor [2,1] contains a pit.

Implementation of the Algorithm

Step.1: Initialize the substitution set to be empty.

Step.2: Recursively unify atomic sentences:

(a) Check for Identical expression match.
(b) If one expression is a variable v_i, and the other is a term t_i which does not contain variable vi, then:
    a. Substitute t_i / v_i in the existing substitutions
    b. Add t_i / v_i to the substitution setlist.
    c. If both the expressions are functions, then function name must be similar, and the number of arguments must be the same in both the expressions.

Unification Exam Problems

For each pair of the following atomic sentences find the most general unifier (If exist).

Problem 1: Find the MGU of {p(f(a), g(Y)) and p(X, X)}

Solution:

S0 => Here, Ψ1 = p(f(a), g(Y)), and Ψ2 = p(X, X)

SUBST θ= {f(a) / X}

S1 => Ψ1 = p(f(a), g(Y)), and Ψ2 = p(f(a), f(a))

SUBST θ= {f(a) / g(y)}, Unification failed.

Unification is not possible for these expressions.

Problem 2: Find the MGU of {p(b, X, f(g(Z))) and p(Z, f(Y), f(Y))}

Solution:

Here, Ψ1 = p(b, X, f(g(Z))) , and Ψ2 = p(Z, f(Y), f(Y))

S0 => { p(b, X, f(g(Z))); p(Z, f(Y), f(Y))}

SUBST θ={b/Z}

S1 => { p(b, X, f(g(b))); p(b, f(Y), f(Y))}

SUBST θ={f(Y) /X}

S2 => { p(b, f(Y), f(g(b))); p(b, f(Y), f(Y))}

SUBST θ= {g(b) /Y}

S2 => { p(b, f(g(b)), f(g(b)); p(b, f(g(b)), f(g(b))} Unified Successfully.

And Unifier = { b/Z, f(Y) /X , g(b) /Y}.

Problem 3: Find the MGU of {p (X, X), and p (Z, f(Z))}

Solution:

Here, Ψ1 = {p (X, X), and Ψ2 = p (Z, f(Z))

S0 => {p (X, X), p (Z, f(Z))}

SUBST θ= {X/Z}

S1 => {p (Z, Z), p (Z, f(Z))}

SUBST θ= {f(Z) / Z}, Unification Failed.

Hence, unification is not possible for these expressions.

Problem 4: Find the MGU of UNIFY(prime (11), prime(y))

Solution:

Here, Ψ1 = {prime(11) , and Ψ2 = prime(y)}

S0 => {prime(11) , prime(y)}

SUBST θ= {11/y}

S1 => {prime(11) , prime(11)} , Successfully unified.

Unifier: {11/y}.

Problem 5: Find the MGU of Q(a, g(x, a), f(y)), Q(a, g(f(b), a), x)}

Solution:

Here, Ψ1 = Q(a, g(x, a), f(y)), and Ψ2 = Q(a, g(f(b), a), x)

S0 => {Q(a, g(x, a), f(y)); Q(a, g(f(b), a), x)}

SUBST θ= {f(b)/x}

S1 => {Q(a, g(f(b), a), f(y)); Q(a, g(f(b), a), f(b))}

SUBST θ= {b/y}

S1 => {Q(a, g(f(b), a), f(b)); Q(a, g(f(b), a), f(b))}, Successfully Unified.

Unifier: [a/a, f(b)/x, b/y].

Problem 6: UNIFY(knows(Richard, x), knows(Richard, John))

Solution:

Here, Ψ1 = knows(Richard, x), and Ψ2 = knows(Richard, John)

S0 => { knows(Richard, x); knows(Richard, John)}

SUBST θ= {John/x}

S1 => { knows(Richard, John); knows(Richard, John)}, Successfully Unified.

Unifier: {John/x}.

Steps for Resolution:

- Conversion of facts into first-order logic.
- Convert FOL statements into CNF
- Negate the statement which needs to prove (proof by contradiction)
- Draw resolution graph (unification).

Problem 1: Knowledge Base

- John likes all kind of food.
- Apple and vegetable are food
- Anything anyone eats and not killed is food.
- Anil eats peanuts and still alive
- Harry eats everything that Anil eats.

Prove by resolution that: John likes peanuts.

Solution:

Step-1: Conversion of Facts into FOL

In the first step we will convert all the given statements into its first order logic.

a. ∀x: food(x) → likes(John, x)

b. food(Apple) Λ food(vegetables)

c. ∀x ∀y: eats(x, y) Λ ¬ killed(x) → food(y)

d. eats (Anil, Peanuts) Λ alive(Anil).

e. ∀x : eats(Anil, x) → eats(Harry, x)

f. ∀x: ¬ killed(x) → alive(x) } **added predicates.**

g. ∀x: alive(x) → ¬ killed(x)

h. likes(John, Peanuts)

Step-2: Conversion of FOL into CNF

In First order logic resolution, it is required to convert the FOL into CNF as CNF form makes easier for resolution proofs.

Eliminate all implication (→) and rewrite

(a) ∀x ¬ food(x) V likes(John, x)

(b) food(Apple) Λ food(vegetables)

(c) ∀x ∀y ¬ [eats(x, y) Λ ¬ killed(x)] V food(y)

(d) eats (Anil, Peanuts) Λ alive(Anil)

(e) ∀x ¬ eats(Anil, x) V eats(Harry, x)

(f) ∀x¬ [¬ killed(x)] V alive(x)

(g) ∀x ¬ alive(x) V ¬ killed(x)

(h) likes(John, Peanuts).

Move negation (¬)inwards and rewrite

(a) ∀x ¬ food(x) V likes(John, x)

(b) food(Apple) Λ food(vegetables)

(c) ∀x ∀y ¬ eats(x, y) V killed(x) V food(y)

(d) eats (Anil, Peanuts) Λ alive(Anil)

(e) ∀x ¬ eats(Anil, x) V eats(Harry, x)

(f) ∀x ¬killed(x)] V alive(x)

(g) ∀x ¬ alive(x) V ¬ killed(x)

(h) likes(John, Peanuts).

Rename variables or standardize variables

(a) ∀x ¬ food(x) V likes(John, x)

(b) food(Apple) Λ food(vegetables)

(c) ∀y ∀z ¬ eats(y, z) V killed(y) V food(z)

(d) eats (Anil, Peanuts) Λ alive(Anil)

(e) ∀w¬ eats(Anil, w) V eats(Harry, w)

(f) ∀g ¬killed(g)] V alive(g)

(g) ∀k ¬ alive(k) V ¬ killed(k)

(h) likes(John, Peanuts).

Eliminate existential instantiation quantifier by elimination.
In this step, we will eliminate existential quantifier ∃, and this process is known as **Skolemization**. But in this example problem since there is no existential quantifier so all the statements will remain same in this step.

Drop Universal quantifiers.
In this step we will drop all universal quantifier since all the statements are not implicitly quantified so we don't need it.

(a) ¬ food(x) V likes(John, x)

(b) food(Apple)

(c) food(vegetables)

(d) ¬ eats(y, z) V killed(y) V food(z)

(e) eats (Anil, Peanuts)

(f) alive(Anil)

(g) ¬ eats(Anil, w) V eats(Harry, w)

(h) killed(g) V alive(g)

(i) ¬ alive(k) V ¬ killed(k)

(j) likes(John, Peanuts).

Note: Statements "food(Apple) Λ food(vegetables)" and "eats (Anil, Peanuts) Λ alive(Anil)" can be written in two separate statements. Distribute conjunction Λ over disjunction ¬.

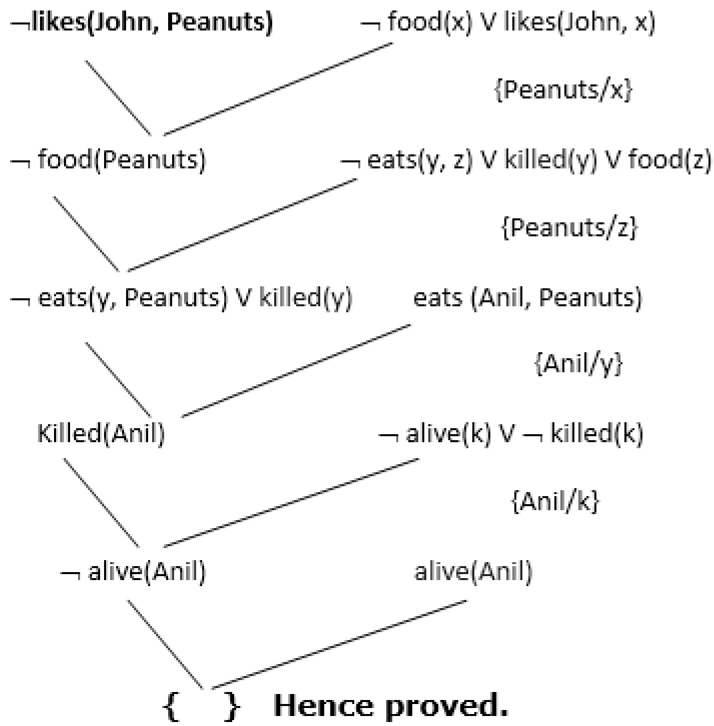
This step will not make any change in this problem.

Step-3: Negate the statement to be proved

In this statement, we will apply negation to the conclusion statements, which will be written as ¬likes(John, Peanuts)

Step-4: Draw Resolution graph:

Now in this step, we will solve the problem by resolution tree using substitution. For the above problem, it will be given as follows:



Hence the negation of the conclusion has been proved as a complete contradiction with the given set of statements.

Explanation of Resolution graph:

- In the first step of resolution graph, **¬likes(John, Peanuts)** , and **likes(John, x)** get resolved(canceled) by substitution of **{Peanuts/x}**, and we are left with **¬ food(Peanuts)**
- In the second step of the resolution graph, **¬ food(Peanuts)** , and **food(z)** get resolved (canceled) by substitution of **{ Peanuts/z}**, and we are left with **¬ eats(y, Peanuts) V killed(y)** .
- In the third step of the resolution graph, **¬ eats(y, Peanuts)** and **eats (Anil, Peanuts)** get resolved by substitution **{Anil/y}**, and we are left with **Killed(Anil)** .
- In the fourth step of the resolution graph, **Killed(Anil)** and **¬ killed(k)** get resolve by substitution **{Anil/k}**, and we are left with **¬ alive(Anil)** .
- In the last step of the resolution graph **¬ alive(Anil)** and **alive(Anil)** get resolved.

Problem 2: Knowledge Base

- All humans are mortal. $\forall x\ (Human(x) \rightarrow Mortal(x))$
- Socrates is a human. Human(Socrates)
- Some philosophers are humans. $\exists x\ (Philosopher(x) \land Human(x))$
- If someone is a philosopher, they are wise. $\forall y\ (Philosopher(y) \rightarrow Wise(y))$
- Socrates is a philosopher. Philosopher(Socrates)
- Wise individuals are knowledgeable. $\forall z\ (Wise(z) \rightarrow Knowledgeable(z))$
- All knowledgeable people are educated. $\forall w\ (Knowledgeable(w) \rightarrow Educated(w))$
- All educated people succeed. $\forall v\ (Educated(v) \rightarrow Succeed(v))$

Goal:

- Prove that Socrates succeeds. Succeed(Socrates)

Solution:

Conversion to CNF (for Predicate Logic).

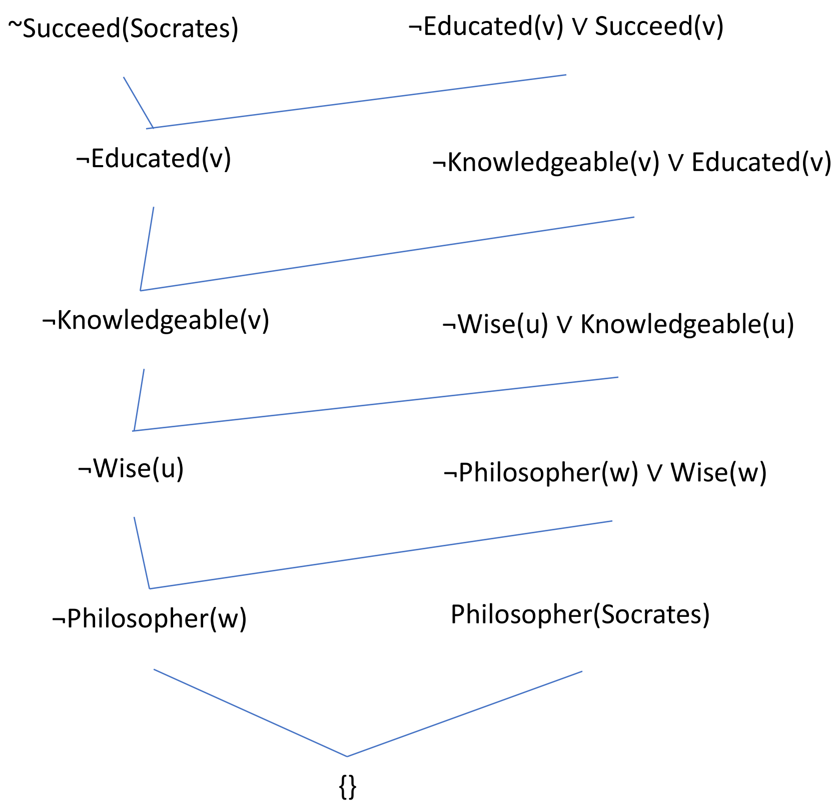
Knowledge Base:

- Clause 1: $\neg Human(x) \lor Mortal(x)$
- Clause 2: Human(Socrates)
- Clause 3: $Philosopher(z) \land Human(z)$
- Clause 4: $\neg Philosopher(w) \lor Wise(w)$
- Clause 5: Philosopher(Socrates)
- Clause 6: $\neg Wise(u) \lor Knowledgeable(u)$
- Clause 7: $\neg Knowledgeable(v) \lor Educated(v)$
- Clause 8: $\neg Educated(v) \lor Succeed(v)$

Goal:

- Goal Clause: Succeed(Socrates)

Draw Resolution graph. Proof by contradiction



Problem 3:

Knowledge Base:

- All students study. ∀x (Student(x) → Study(x))
- Jane is a student. Student(Jane)
- Some students study mathematics. ∃y (Student(y) ∧ Mathematics(y))
- All students studying mathematics take calculus. ∀z (Mathematics(z) → Takes(z, Calculus))
- Jane studies mathematics. Mathematics(Jane)
- Calculus is a course. Course(Calculus)
- All students who take calculus are serious students. ∀w (Takes(w, Calculus) → Serious(w))
- Serious students succeed. ∀v (Serious(v) → Succeed(v))

Goal:

- Prove that Jane succeeds. Succeed(Jane)

Solution:

Conversion to CNF (for Predicate Logic):

Converting the statements to CNF:

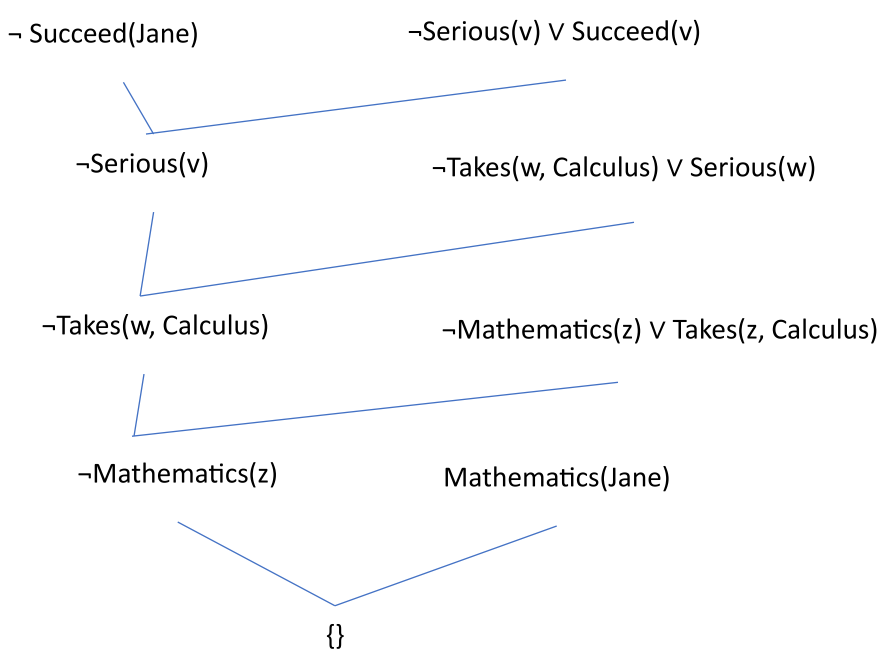
Knowledge Base:

- Clause 1: ¬Student(x) ∨ Study(x)
- Clause 2: Student(Jane)
- Clause 3: Student(y) ∧ Mathematics(y)
- Clause 4: ¬Mathematics(z) ∨ Takes(z, Calculus)
- Clause 5: Mathematics(Jane)
- Clause 6: Course(Calculus)
- Clause 7: ¬Takes(w, Calculus) ∨ Serious(w)
- Clause 8: ¬Serious(v) ∨ Succeed(v)

Goal:

Goal Clause: Succeed(Jane)

Solution:



Problem 4:

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.

Prove that Colonel West is a Criminal

Solution:

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

R1:American(x) $\wedge$ Weapon(y) $\wedge$ Sells(x,y,z) $\wedge$ Hostile(z) $\Rightarrow$ Criminal(x)

Nono … has some missiles, i.e.,

$\exists$x Owns(Nono,x) $\wedge$ Missile(x):

R2: Owns(Nono,M_1) and Missile(M_1)

R3: Missile(M1)

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

R4: Missile(x) $\wedge$ Owns(Nono,x) $\Rightarrow$ Sells(West,x,Nono)

Missiles are weapons:

R5: Missile(x) $\Rightarrow$ Weapon(x)

An enemy of America counts as "hostile“:

R6: Enemy(x,America) $\Rightarrow$ Hostile(x)

West, who is American …

R7: American(West)

The country Nono, an enemy of America …

R8: Enemy(Nono,America)

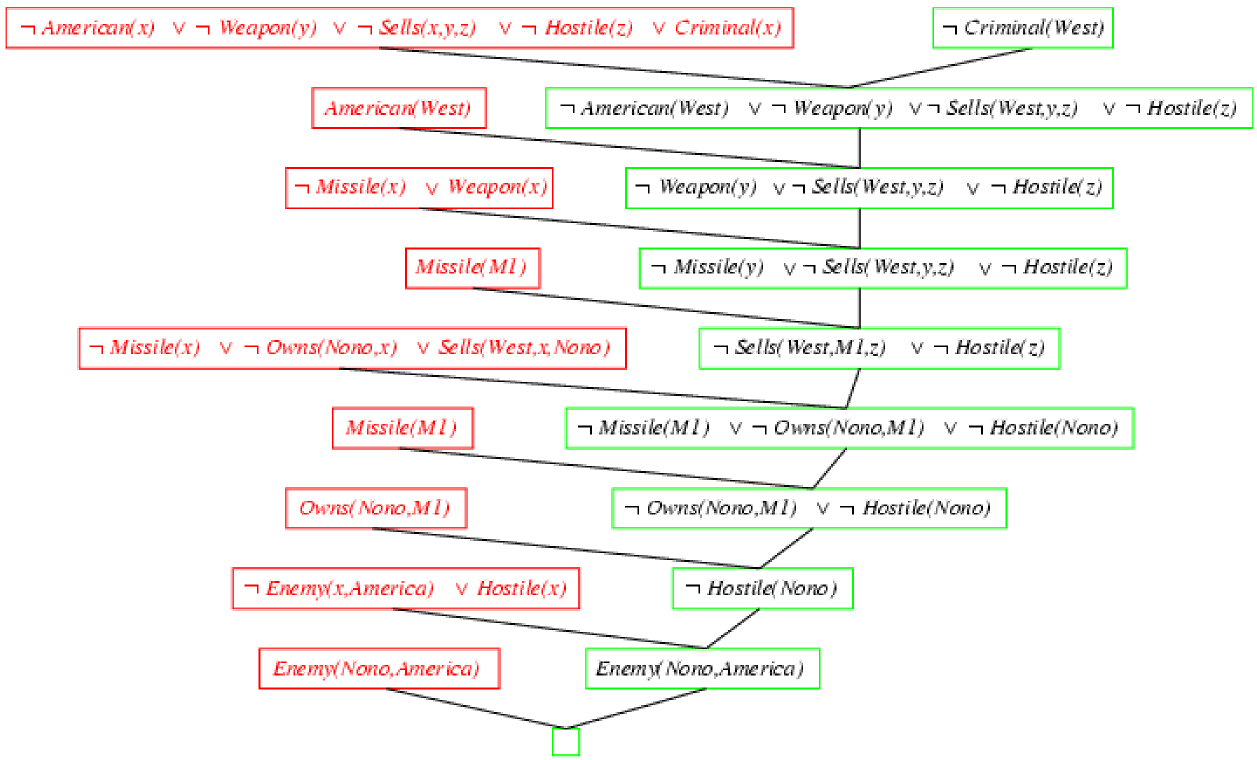
Convert to CNF

- ¬American(x) ∨ ¬Weapon(y) ∨ ¬Sells(x, y, z) ∨ ¬Hostile(z) ∨ Criminal(x)
- ¬Missile(x) ∨ ¬Owns(Nono, x) ∨ Sells(West, x, Nono)
- ¬Enemy(x, America) ∨ Hostile(x)
- ¬Missile(x) ∨ Weapon(x)
- Owns(Nono, M1)
- Missile(M1)
- American(West)

- Enemy(Nono, America) .

Query: Criminal(West)

Proof by Contradiction:

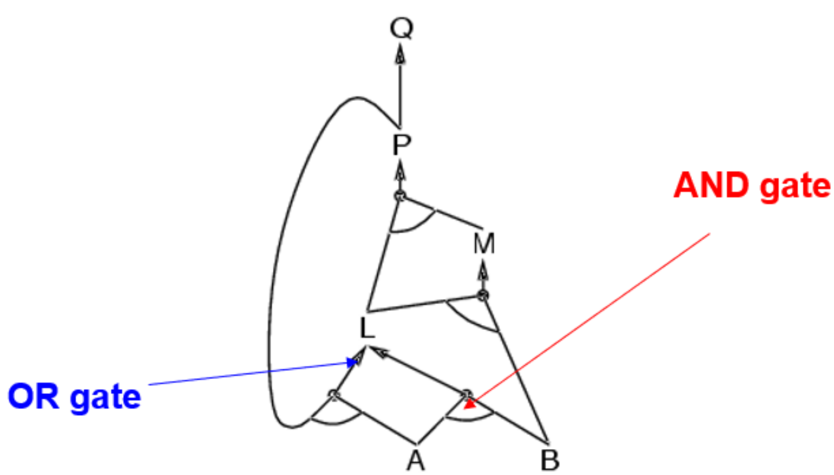


Forward and backward Problem for Propositional Logic

Problem:

$$
\begin{aligned}
P &\Rightarrow Q \\
L \wedge M &\Rightarrow P \\
B \wedge L &\Rightarrow M \\
A \wedge P &\Rightarrow L \\
A \wedge B &\Rightarrow L \\
A & \\
B &
\end{aligned}
$$

Solution:



Forward and backward Problem for FOL

Problem

“As per the law, it is a crime for an American to sell weapons to hostile nations. Country A, an enemy of America, has some missiles, and all the missiles were sold to it by Robert, who is an American citizen."

Prove that **"Robert is criminal."**

To solve the above problem, first, we will convert all the above facts into first-order definite clauses, and then we will use a forward-chaining algorithm to reach the goal.

Facts Conversion into FOL:

It is a crime for an American to sell weapons to hostile nations. (Let's say p, q, and r are variables)
American (p) ∧ weapon(q) ∧ sells (p, q, r) ∧ hostile(r) → Criminal(p) ...(1)

Country A has some missiles. **Owns(A, p) ∧ Missile(p)**. It can be written in two definite clauses by using Existential Instantiation, introducing new Constant T1.
Owns(A, T1) ......(2)
Missile(T1) ....(3)

All of the missiles were sold to country A by Robert.
Missiles(p) ∧ Owns (A, p) → Sells (Robert, p, A) ......(4)

Missiles are weapons.
Missile(p) → Weapons (p) ....(5)

Enemy of America is known as hostile.
Enemy(p, America) →Hostile(p) ....(6)

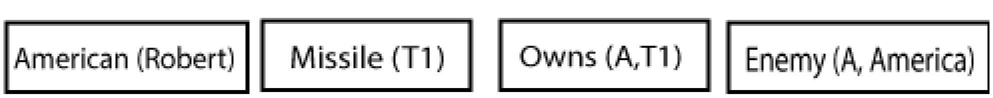
Country A is an enemy of America.
Enemy (A, America) ....(7)

Robert is American
American(Robert). ....(8)

Forward chaining proof

Step-1:

In the first step we will start with the known facts and will choose the sentences which do not have implications, such as: **American(Robert), Enemy(A, America), Owns(A, T1), and Missile(T1)**. All these facts will be represented below.



Step-2:

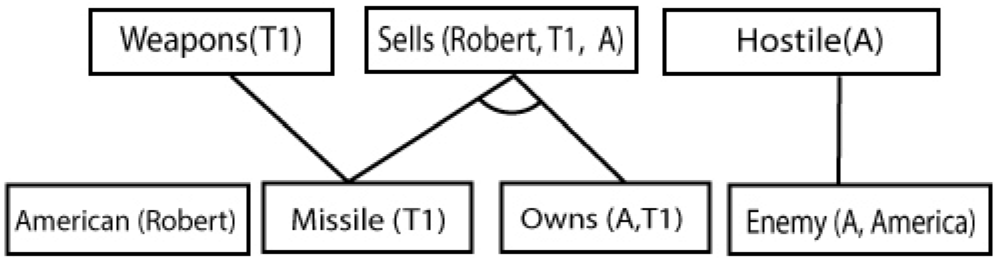
At the second step, we will see those facts which infer from available facts and with satisfied premises.

Rule-(1) does not satisfy premises, so it will not be added in the first iteration.

Rule-(2) and (3) are already added.

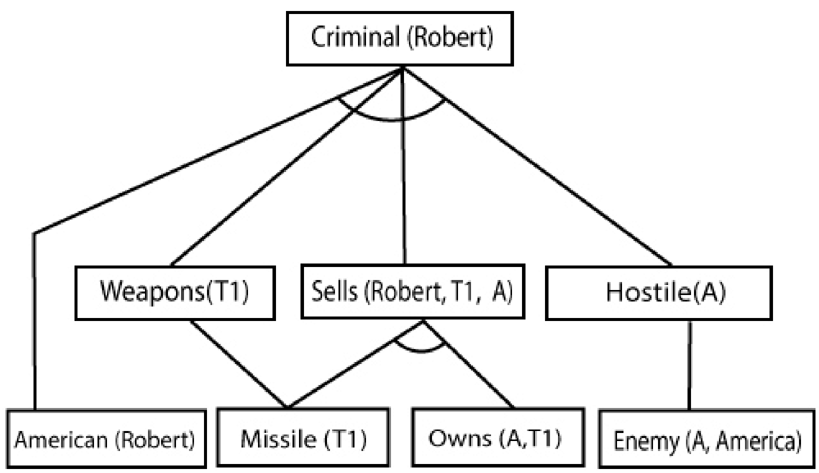
Rule-(4) satisfy with the substitution {p/T1}, **so Sells (Robert, T1, A)** is added, which infers from the conjunction of Rule (2) and (3).

Rule-(6) is satisfied with the substitution(p/A), so Hostile(A) is added and which infers from Rule-(7).



Step-3:

At step-3, as we can check Rule-(1) is satisfied with the substitution **{p/Robert, q/T1, r/A}, so we can add Criminal(Robert)** which infers all the available facts. And hence we reached our goal statement.



Hence it is proved that Robert is Criminal using forward chaining approach.

Backward Chaining:

Backward-chaining is also known as a backward deduction or backward reasoning method when using an inference engine. A backward chaining algorithm is a form of reasoning, which starts with the goal and works backward, chaining through rules to find known facts that support the goal.

Properties of backward chaining:

It is known as a top-down approach.

Backward-chaining is based on modus ponens inference rule.

In backward chaining, the goal is broken into sub-goal or sub-goals to prove the facts true.

It is called a goal-driven approach, as a list of goals decides which rules are selected and used.

Backward -chaining algorithm is used in game theory, automated theorem proving tools, inference engines, proof assistants, and various AI applications.

The backward-chaining method mostly used a **depth-first search** strategy for proof.

Example:

In backward chaining, we will use the same above example, and will rewrite all the rules.

American (p) ∧ weapon(q) ∧ sells (p, q, r) ∧ hostile(r) → Criminal(p) ...(1)
Owns(A, T1) ....(2)

Missile(T1) ....(3)

Missiles(p) ∧ Owns (A, p) → Sells (Robert, p, A) ......(4)

Missile(p) → Weapons (p) ....(5)

Enemy(p, America) →Hostile(p) ....(6)

Enemy (A, America) ....(7)

American(Robert) ....(8)

Backward-Chaining proof:

In Backward chaining, we will start with our goal predicate, which is **Criminal(Robert)**, and then infer further rules.

Step-1:

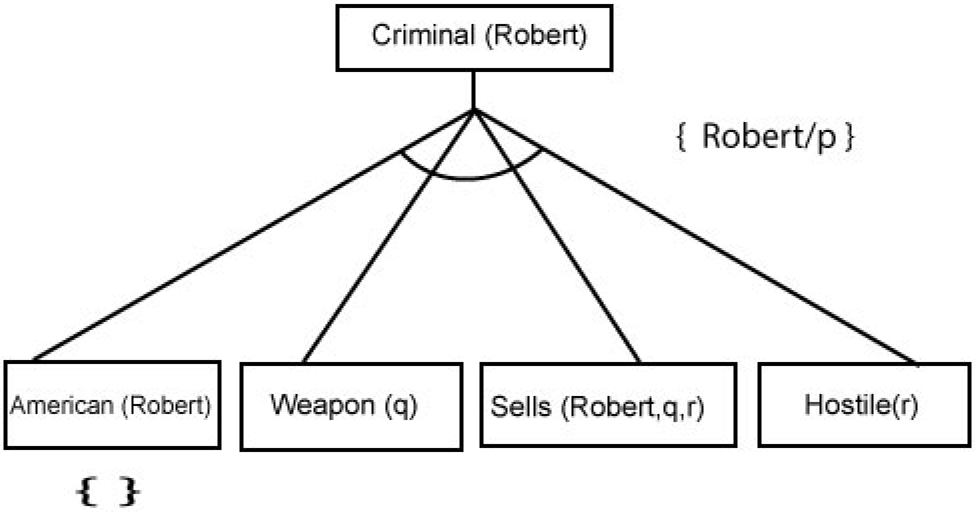
At the first step, we will take the goal fact. And from the goal fact, we will infer other facts, and at last, we will prove those facts true. So our goal fact is "Robert is Criminal," so following is the predicate of it.

Criminal (Robert)

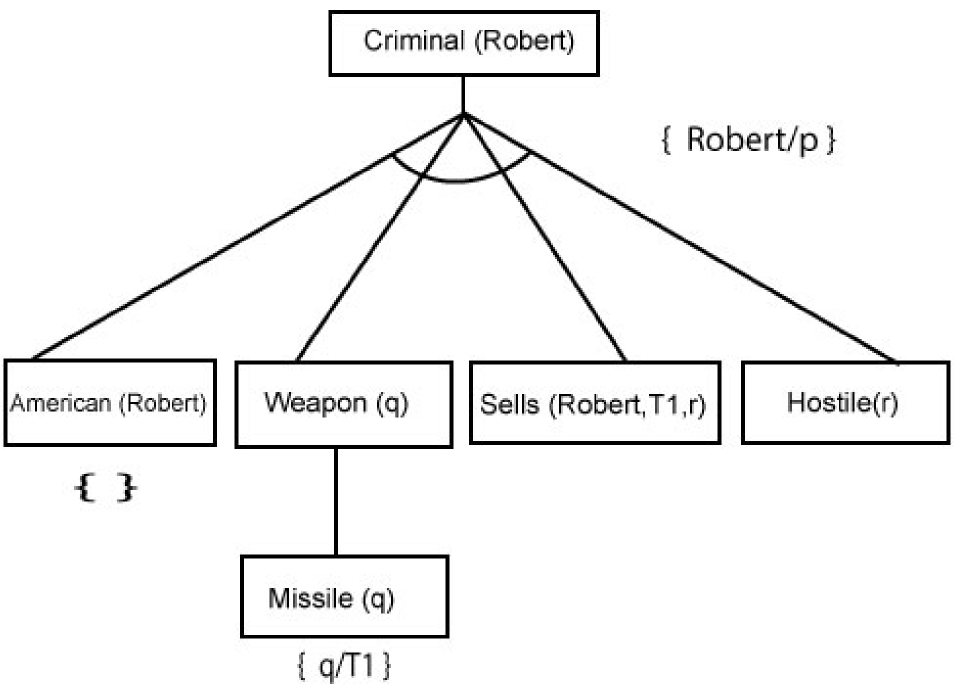
Step-2:

At the second step, we will infer other facts form goal fact which satisfies the rules. So as we can see in Rule-1, the goal predicate Criminal (Robert) is present with substitution {Robert/P}. So we will add all the conjunctive facts below the first level and will replace p with Robert.

Here we can see American (Robert) is a fact, so it is proved here.

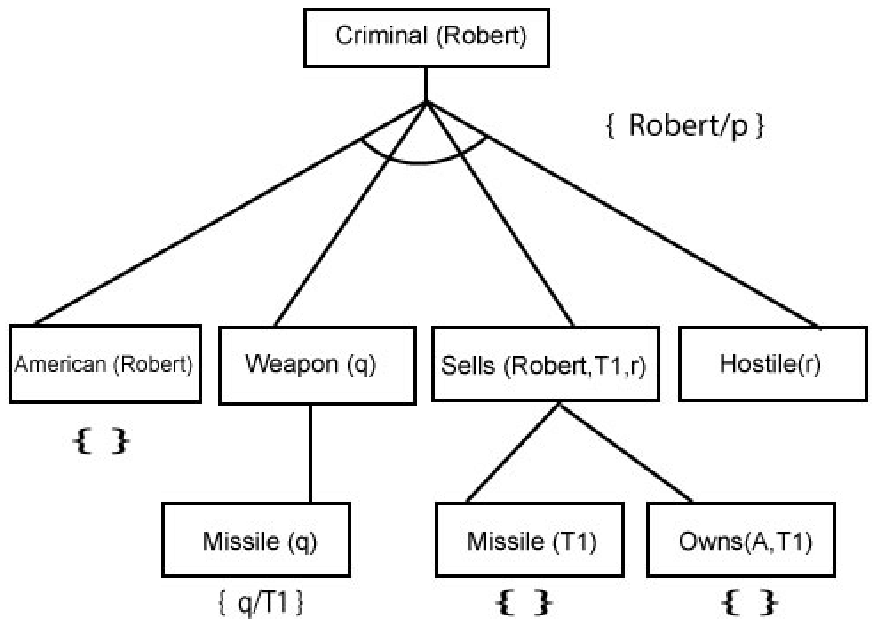


Step-3:t At step-3, we will extract further fact Missile(q) which infer from Weapon(q), as it satisfies Rule-(5). Weapon (q) is also true with the substitution of a constant T1 at q.



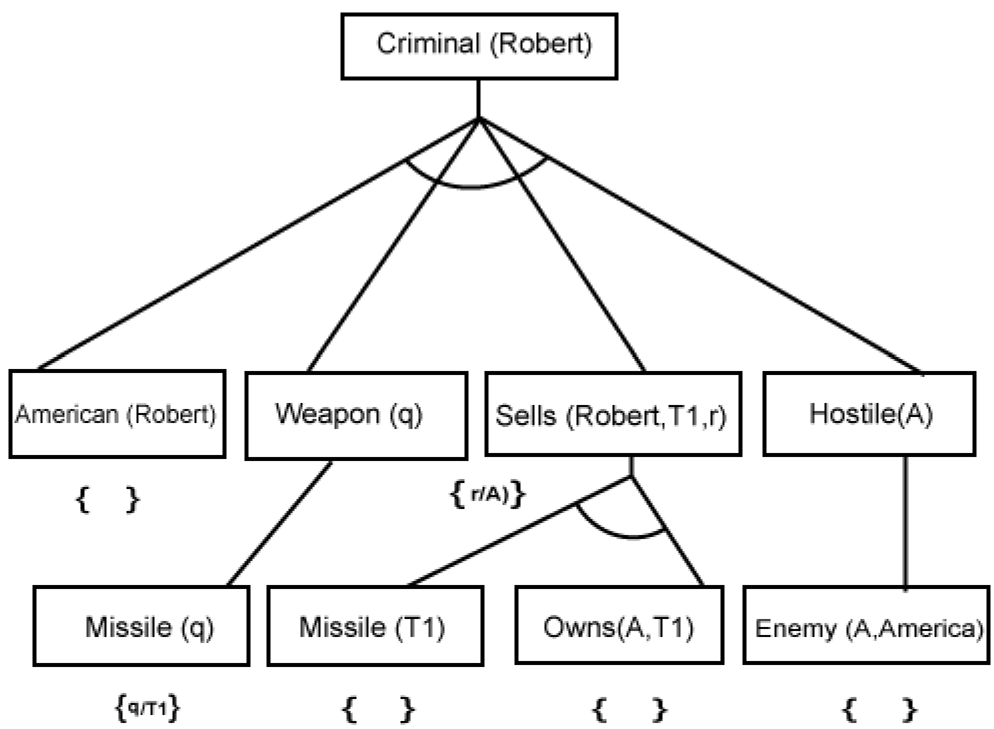
Step-4:

At step-4, we can infer facts Missile(T1) and Owns(A, T1) form Sells(Robert, T1, r) which satisfies the **Rule- 4**, with the substitution of A in place of r. So these two statements are proved here.



Step-5:

At step-5, we can infer the fact **Enemy(A, America)** from **Hostile(A)** which satisfies Rule- 6. And hence all the statements are proved true using backward chaining.



Simple Bayes Rule's Problem(Exam Problem)

Problem 1: A doctor is aware that disease meningitis causes a patient to have a stiff neck, and it occurs 80% of the time. He is also aware of some more facts, which are given as follows:

- The Known probability that a patient has meningitis disease is 1/30,000.
- The Known probability that a patient has a stiff neck is 2%.

what is the probability that a patient has diseases meningitis with a stiff neck?

Solution:

A doctor is aware that disease meningitis causes a patient to have a stiff neck, and it occurs 80% of the time. He is also aware of some more facts, which are given as follows:

The Known probability that a patient has meningitis disease is 1/30,000.

The Known probability that a patient has a stiff neck is 2%.

Let a be the proposition that patient has stiff neck and b be the proposition that patient has meningitis. , so we can calculate the following as:

P(a|b) = 0.8

P(b) = 1/30000

P(a)= .02

$$\mathbf{P(b|a)} = \frac{P(a|b)P(b)}{P(a)} = \frac{0.8*(\frac{1}{30000})}{0.02} = 0.001333333.$$

Problem 2: From a standard deck of playing cards, a single card is drawn. The probability that the card is king is 4/52, then calculate posterior probability P(King|Face), which means the drawn face card is a king card.

Solution:

$$\mathbf{P(king|face)} = \frac{P(Face|king)*P(King)}{P(Face)} \quad ....(i)$$

(king): probability that the card is King= 4/52= 1/13

P(face): probability that a card is a face card= 3/13

P(Face|King): probability of face card when we assume it is a king = 1

Putting all values in equation (i) we will get:

$$P(\text{king}|\text{face}) = \frac{1 * (\frac{1}{13})}{(\frac{3}{13})}$$ **= 1/3, it is a probability that a face card is a king card.**

Problem 3: Let's take System *Y*, which has a malfunction rate of 0.5%. There is a diagnostic test available for detecting malfunctions in System *Y*. It correctly identifies 98% of System *Y's* malfunctions. Also, the test has a false positive rate of 3%. What is the probability of System *Y* having a malfunction given that the test result is positive?

Solution:

We can apply Bayes' theorem to calculate $P(Malfunction|Positivetest)$.
$P(Malfunction|Positivetest)=(P(Malfunction)*P(Positivetest|Malfunction))/P(Positivetest)$

$P(Malfunction)=0.5\%=0.005$

$P(Positivetest|Malfunction)=0.98$

$P(NoMalfunction)=100\%-0.5\%$

$P(NoMalfunction)=99.5\%$

$P(Positivetest|NoMalfunction)=0.03$

Since we don't have P(Positive Test), we'll have to calculate it using the rest of the values. For this, we need to consider the probabilities of getting a positive test result in both malfunction and non-malfunction cases.
$P(Positivetest)=(P(Malfunction)*P(Positivetest|Malfunction))+(P(NoMalfunction)*P(Positivetest|NoMalfunction))$

$P(PositiveTest)=(0.005*0.98)+(0.995*0.03)$

$P(PositiveTest)=0.035$

Plugging in the values in Bayes' theorem gives us our result:

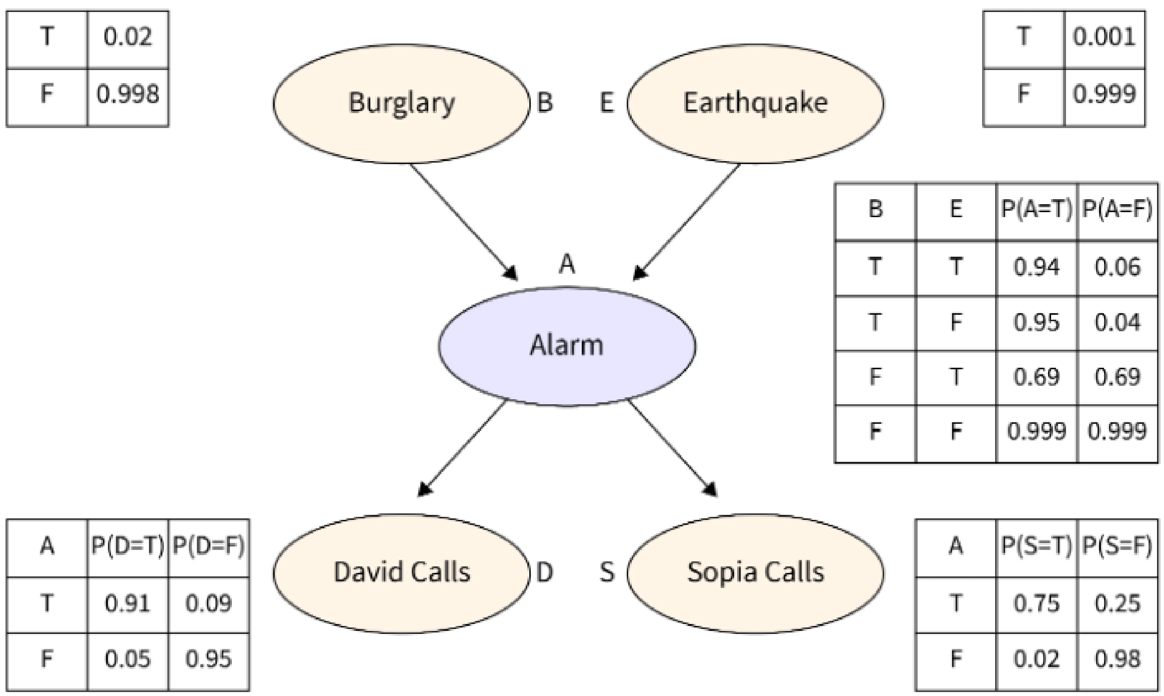
$P(Malfunction|PositiveTest)=(0.005*0.98)/0.035$

$P(Malfunction|PositiveTest)=0.142857.$

Example of Bayesian Network

Problem 1: To prevent break-ins, Harry put a brand-new burglar alarm at his house. The alarm consistently reacts to a break-in, but it also reacts to little earthquakes. James and Safina, two of Harry's neighbours, have agreed to call Harry at work when they hear the alarm. James always calls Harry when the alarm goes off, but occasionally he gets distracted by the phone ringing and calls at other times. Safina, on the other hand, enjoys listening to loud music, so occasionally she doesn't hear the alarm. Here, we'd want to calculate the likelihood of a burglar alarm.

Determine the likelihood that the alarm went off but that neither a burglary nor an earthquake had taken place, and that both James and Safina had phoned Harry.



Solution:

List of All Events Occurring in a Bayesian Network

- Burglary (B)
- Earthquake(E)
- Alarm(A)
- James Calls(D)
- Safina calls(S)

Let's take the observed probability for the Burglary and earthquake component:

- $P(B{=}True){=}0.002$, which is the probability of burglary.
- $P(B{=}False){=}0.998$, which is the probability of no burglary.

- $P(E=True)=0.001$, which is the probability of a minor earthquake
- $P(E=False)=0.999$, Which is the probability that an earthquake not occurred.

From the formula of joint distribution, we can write the problem statement in the form of probability distribution:

*P(S, D, A, ¬B, ¬E) = P (S|A) *P (D|A)*P (A|¬B ^ ¬E) *P (¬B) *P (¬E).*

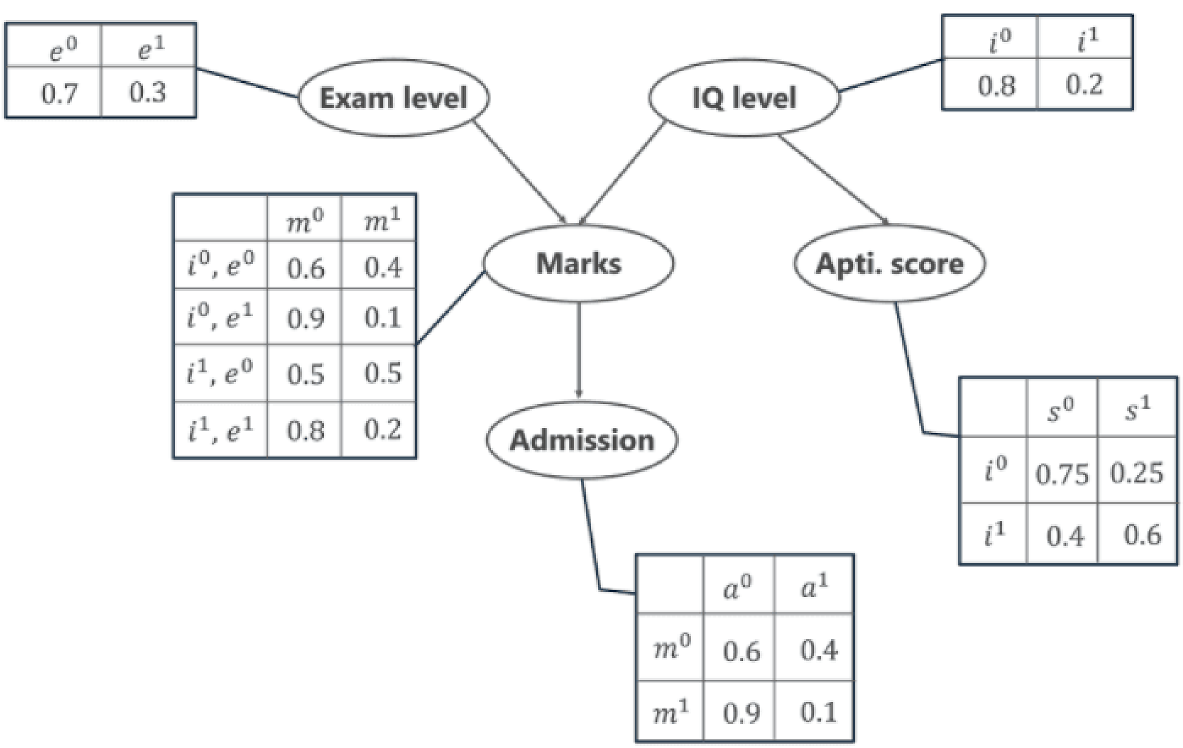
= 0.75 0.91* 0.001* 0.998*0.999*

= 0.00068045.

Problem 2: Let us now understand the mechanism of Bayesian Networks and their advantages with the help of a simple example. In this example, let us imagine that we are given the task of modeling a student's marks (m) for an exam he has just given. From the given Bayesian Network Graph below, we see that the marks depend upon two other variables. They are,

- Exam Level (e)– This discrete variable denotes the difficulty of the exam and has two values (0 for easy and 1 for difficult)
- IQ Level (i) – This represents the Intelligence Quotient level of the student and is also discrete in nature having two values (0 for low and 1 for high)

Additionally, the IQ level of the student also leads us to another variable, which is the Aptitude Score of the student (s). Now, with marks the student has scored, he can secure admission to a particular university. The probability distribution for getting admitted (a) to a university is also given below.



Case 1: Calculate the probability that in spite of the exam level being difficult, the student having a low IQ level and a low Aptitude Score, manages to pass the exam and secure admission to the university.

Case 2: In another case, calculate the probability that the student has a High IQ level and Aptitude Score, the exam being easy yet fails to pass and does not secure admission to the university.

Solution:

In the above graph, we see several tables representing the probability distribution values of the given 5 variables. These tables are called the Conditional Probabilities Table or CPT. There are a few properties of the CPT given below –

- The sum of the CPT values in each row must be equal to 1 because all the possible cases for a particular variable are exhaustive (representing all possibilities).
- If a variable that is Boolean in nature has k Boolean parents, then in the CPT it has 2K probability values.

Coming back to our problem, let us first list all the possible events that are occurring in the above-given table.

1. Exam Level (e)
2. IQ Level (i)
3. Aptitude Score (s)
4. Marks (m)
5. Admission (a)

These five variables are represented in the form of a Directed Acyclic Graph (DAG) in a Bayesian Network format with their Conditional Probability tables. Now, to calculate the Joint Probability Distribution of the 5 variables the formula is given by,

P[a, m, i, e, s]= P(a | m) . P(m | i, e) . P(i) . P(e) . P(s | i)

From the above formula,

- P(a | m) denotes the conditional probability of the student getting admission based on the marks he has scored in the examination.
- P(m | i, e) represents the marks that the student will score given his IQ level and difficulty of the Exam Level.
- P(i) and P(e) represent the probability of the IQ Level and the Exam Level.
- P(s | i) is the conditional probability of the student's Aptitude Score, given his IQ Level.

With the following probabilities calculated, we can find the Joint Probability Distribution of the entire Bayesian Network.

Case 1:

From the above word problem statement, the Joint Probability Distribution can be written as below,

P[a=1, m=1, i=0, e=1, s=0]

From the above Conditional Probability tables, the values for the given conditions are fed to the formula and is calculated as below.

P[a=1, m=1, i=0, e=0, s=0] = P(a=1 | m=1) . P(m=1 | i=0, e=1) . P(i=0) . P(e=1) . P(s=0 | i=0)

= 0.1 * 0.1 * 0.8 * 0.3 * 0.75

= 0.0018

Case 2:

The formula for the JPD is given by

P[a=0, m=0, i=1, e=0, s=1]

Thus,

P[a=0, m=0, i=1, e=0, s=1]= P(a=0 | m=0) . P(m=0 | i=1, e=0) . P(i=1) . P(e=0) . P(s=1 | i=1)

= 0.6 * 0.5 * 0.2 * 0.7 * 0.6

= 0.0252