National University of Singapore
School of Computing
IT5005 Artificial Intelligence

Propositional Logic 2

1. Use resolution to show that the statement

$$(P \lor Q) \land (\neg P \lor R) \land (\neg P \lor \neg R) \land (P \lor \neg Q)$$

is UNSAT.

Solution:

- $L_1 : P \lor Q$ (given)
- $L_2 : \neg P \lor R$ (given)
- $L_3 : \neg P \lor \neg R$ (given)
- $L_4 : P \lor \neg Q$ (given)
- $L_5 : P$ (L_1, L_4, resolution)
- $L_6 : R$ (L_2, L_5, resolution)
- $L_7 : \neg R$ (L_3, L_5, resolution)
- $L_8 :$ false (L_6, L_7, resolution)

2. Suppose we are given the following premises:

- $P_1 : P \Rightarrow Q$
- $P_2 : R \Rightarrow P$
- $P_3 : \neg Q$
- $P_4 : R \lor P \lor S$

Use resolution to prove that S is always True under the premises.

Solution:

- $L_1 : \neg P \lor Q$ (given)

- $L_2 : \neg R \vee P$ (given)
- $L_3 : \neg Q$ (premise)
- $L_4 : R \vee P \vee S$ (given)
- $L_5 : \neg S$ (negation of conclusion)
- $L_6 : R \vee P$ (L_4, L_5, resolution)
- $L_7 : P$ (L_2, L_6, resolution)
- $L_8 : Q$ (L_1, L_7, resolution)
- L_9 : false (L_3, L_8, resolution)

3. Consider the following Horn clauses:

   - $P_1 : Rain$
   - $P_2 : Weekday$
   - $P_3 : Rain \Rightarrow Wet$
   - $P_4 : Wet \wedge Weekday \Rightarrow Traffic$
   - $P_5 : Traffic \wedge Careless \Rightarrow Accident$

   Prove $Traffic$ with both forward and backward chaining algorithms

   Could you prove $Accident$ for the given KB?

   Solution:

   The AND-OR graph of the KB is shown in Fig. 1:

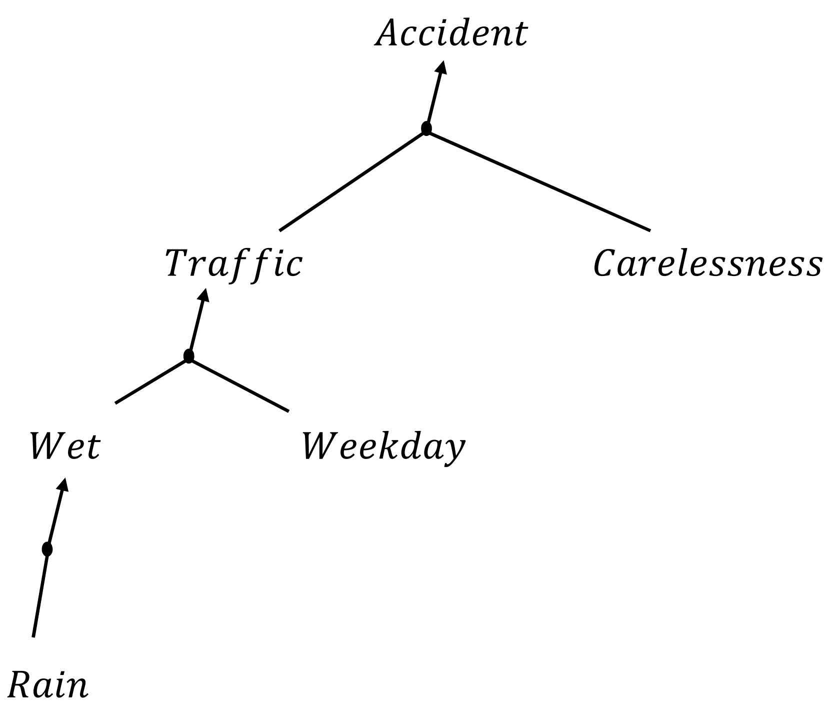

Figure 1: AND-OR Grap

13

Clause	c	count(c) = # of antecedents of c
$Rain$	1	0
$Weekday$	2	0
$Rain \Rightarrow Wet$	3	1
$Wet \wedge Weekday \Rightarrow Traffic$	4	2
$Traffic \wedge Careless \Rightarrow Accident$	5	2

Query: $Traffic$

Queue	p=Pop(Queue)	p=Query	Remark
$[Rain, Weekday]$	$Rain$	No	$Rain$ is already in KB; inferred($Rain$) = True; count(3) = 1-1 = 0; so we can derive $Wet = True$ Add Wet to queue
$[Weekday, Wet]$	$Weekday$	No	$Weekday$ is already in KB; inferred($Weekday$) = True; count(4) = 2-1=1
$[Wet]$	Wet	No	Inferred(Wet) = True count(4) = 0; so we can derive $Traffic = True$ add $Traffic$ to queue
$[Traffic]$	$Traffic$	Yes	Return $Traffic = True$

Query: $Accident$

Queue	p=Pop(Queue)	p=Query	Remark
$[Rain, Weekday]$	$Rain$	No	$Rain$ is already in KB; inferred($Rain$) = True; count(3) = 1-1 = 0; so we can derive $Wet = True$ Add Wet to queue
$[Weekday, Wet]$	$Weekday$	No	$Weekday$ is already in KB; inferred($Weekday$) = True; count(4) = 2-1=1
$[Wet]$	Wet	No	Inferred(Wet) = True count(4) = 0; so we can derive $Traffic = True$ add $Traffic$ to queue
$[Traffic]$	$Traffic$	Yes	Inferred$[Traffic] = True$ count(5) = 1
[]			Queue is empty and return Accident $= False$

Figure 2: Trace of Forward Chaining Algorithm

The query $Accident$ is not a fact
The query $Accident$ is a consequent in P_5.
To prove $Accident$, both $Traffic$ and $Carelessness$ needs to be proved,
i.e., $count(Accident) = 2$;
Make recursive calls to check whether $Traffic$ and $Carelessness$ are true
Recursive call to check $Traffic$:
    returns $Traffic = True$
    (Trace is shown in the previous question)
add Traffic to KB
count($Accident$) = 2-1 = 1
Recursive call to check $Carelessness$:
    The query $Carelessness$is not a fact
    There is no premise with consequent that matches the query $Carelessness$;
    return $Carelessness = False$
Returns $Accident$ = False

Figure 3: Trace of Backward Chaining Algorithm for the query $Traffic$

21

The query $Accident$ is not a fact
The query $Accident$ is a consequent in P_5.
To prove $Accident$, both $Traffic$ and $Carelessness$ needs to be proved,
i.e., $count(Accident) = 2$;
Make recursive calls to check whether $Traffic$ and $Carelessness$ are true
Recursive call to check $Traffic$:
    returns $Traffic = True$
    (Trace is shown in the previous question)
add Traffic to KB
count($Accident$) = 2-1 = 1
Recursive call to check $Carelessness$:
    The query $Carelessness$is not a fact
    There is no premise with consequent that matches the query $Carelessness$;
    return $Carelessness = False$
Returns $Accident$ = False

Figure 4: Trace of Backward Chaining Algorithm for the query $Accident$

21

4. Consider the following knowledge base (KB) with propositional symbols A, B, C, and D. The KB contains two sentences $\mathbf{P1}$ and $\mathbf{P2}$:

$$\begin{aligned} \mathbf{P1} &: C \vee D \\ \mathbf{P2} &: B \Rightarrow ((A \wedge B) \Rightarrow C) \end{aligned}$$

Identify the models of the sentence $\mathbf{P2}$.

Solution:

A	B	C	$(A \wedge B) \Rightarrow C$	$B \Rightarrow ((A \wedge B) \Rightarrow C)$
true	true	true	true	true
true	true	false	false	false
true	false	true	true	true
true	false	false	true	true
false	true	true	true	true
false	true	false	true	true
false	false	true	true	true
false	false	false	true	true

There are 14 models: except

$\{A = true, B = true, C = false, D = true\}$

and

$\{A = true, B = true, C = false, D = false\}$,

all other assignments to the variables A, B, C, and D are models of the sentence $\mathbf{P2}$.

5. Using logic, you need to determine whether the following statement is true: "The vase was broken". The following clues are given:

- **R1**: Charlie was outside.
- **R2**: The vase was broken if and only if the cat was in the house or Bob was playing indoors.
- **R3**: If Bob was playing indoors, then Charlie was outside.
- **R4**: If Charlie was outside, then cat was in the house.

Use the propositional symbols shown in Table 1 to represent the sentences in the puzzle. Answer the following questions:

Table 1: Propositional Symbols

O: Charlie was outside.	**B**: The vase was broken.
H: The cat was in the house.	**I**: Bob was playing indoors.

(a) Translate the clues **R1** to **R4** into propositional form.

(b) Convert the above sentences in propositional form to CNF.

(c) Check whether the statement "The vase was broken" is true or not using resolution-refutation algorithm.

Solution:

(a) Propositional Form

- **R1**: O
- **R2**: $B \iff H \vee I$
- **R3**: $I \Rightarrow O$
- **R4**: $O \Rightarrow H$

(b) Conjunctive Normal Form (CNF)

- **R1**: O
- **R2**: $(\neg B \vee H \vee I) \wedge (\neg H \vee B) \wedge (\neg I \vee B)$
- **R3**: $\neg I \vee O$
- **R4**: $\neg O \vee H$

(c) Resolution-Refutation

Query: The vase is broken: B

- **R1**: O (given)
- **R2a**: $(\neg B \vee H \vee I)$. (given)
- **R2b**: $(\neg H \vee B)$. (given)
- **R2c**: $(\neg I \vee B)$ (given)
- **R3**: $\neg I \vee O$ (given)
- **R4**: $\neg O \vee H$ (given)
- **R5**: $\neg B$. (Negation of Query)
- **R6**: H. (Resolution of **R1** and **R4**)
- **R7**: $\neg H$. (Resolution of **R5** and **R2b**)
- **R8**: Contradiction (from **R6** and **R7**)