Artificial Intelligence

Lecturer 8-10 – Propositional Logic

Lecturers :
Assoc. Prof. **Dr. Hai V. Pham**
Assoc. Prof. Dr. **Le Thanh Huong**
HUST

https://users.soict.hust.edu.vn/haipv

Outline

- What is Logic
- Propositional Logic
  - Syntax
  - Semantic
- Inference in Propositional Logic
  - Forward Chaining
  - Backward Chaining
  - Resolution

Knowledge-based Agents

- Know about the world
  - They maintain a collection of facts (sentences) about the world, their Knowledge Base, expressed in some formal language.
- Reason about the world
  - They are able to derive new facts from those in the KB using some inference mechanism.
- Act upon the world
  - They map percepts to actions by querying and updating the KB.

What is Logic ?

- A logic is a triplet <L,S,R>
  - L, the language of the logic, is a class of sentences described by a precise syntax, usually a formal grammar
  - S, the logic's semantic, describes the meaning of elements in L
  - R, the logic's inference system, consisting of derivation rules over L
- Examples of logics:
  - Propositional, First Order, Higher Order, Temporal, Fuzzy, Modal, Linear, …

Propositional Logic

- Propositional Logic is about facts in the world that are either true or false, nothing else
- Propositional variables stand for basic facts
- Sentences are made of
  - propositional variables (A,B,…),
  - logical constants (TRUE, FALSE), and
  - logical connectives (not,and,or,..)
- The meaning of sentences ranges over the Boolean values {True, False}
  - Examples: It's sunny, John is married

Language of Propositional Logic

- Symbols
  - Propositional variables: A,B,…,P,Q,…
  - Logical constants: TRUE, FALSE
  - Logical connectives: $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$
- Sentences
  - Each propositional variable is a sentence
  - Each logical constant is a sentence
  - If α and β are sentences then the following are sentences

$$(\alpha), \neg\alpha, \alpha \wedge \beta, \alpha \vee \beta, \alpha \Rightarrow \beta, \alpha \Leftrightarrow \beta$$

Formal Language of Propositional Logic

- Symbols
  - Propositional variables: A,B,…,P,Q,…
  - Logical constants: T, F
  - Logical connectives: ¬ , ∧ , ∨ , ⇒ , ⇔
- Formal Grammar
  - Sentence -> Asentence | Csentence
  - Asentence -> TRUE | FALSE | A | B|…
  - Csentence -> (Sentence) | ¬Sentence | Sentence Connective Sentence
  - Connective -> ¬ | ∧ | ∨ |⇒ |⇔

Semantic of Propositional Logic

- The meaning of TRUE is always True, the meaning of FALSE is always False
- The meaning of a propositional variable is either True or False
  - depends on the interpretation
    - assignment of Boolean values to propositional variables
- The meaning of a sentence is either True or False
  - depends on the interpretation

Propositional logic: Syntax

- Propositional logic is the simplest logic – illustrates basic ideas
- The proposition symbols P_1, P_2 etc are sentences
  - If S is a sentence, ¬S is a sentence (negation)
  - If S_1 and S_2 are sentences, $S_1 \wedge S_2$ is a sentence (conjunction)
  - If S_1 and S_2 are sentences, $S_1 \vee S_2$ is a sentence (disjunction)
  - If S_1 and S_2 are sentences, $S_1 \Rightarrow S_2$ is a sentence (implication)
  - If S_1 and S_2 are sentences, $S_1 \Leftrightarrow S_2$ is a sentence (biconditional)

Semantic of Propositional Logic

- True table

P	Q	Not P	P and Q	P or Q	P implies Q	P equiv Q
False	False	True	False	False	True	True
False	True	True	False	True	True	False
True	False	False	False	True	False	False
True	True	False	True	True	True	True

$a \Rightarrow b \Leftrightarrow \neg a \vee b \Leftrightarrow \neg b \Rightarrow \neg a$

Semantic of Propositional Logic

- Satisfiability
  - A sentence is satisfiable if it is true under some interpretation
  - Ex: P or H is satisfiable
    P and ¬P is unsatisfiable (not satisfiable)
- Validity
  - A sentence is valid if it is true in every interpretation
  - Ex: ((P or H) and ¬ A) => P is valid
    P or H is not valid

Semantic of Propositional Logic

- Entailment
  - Given
    - A set of sentences Γ
    - A sentence ψ
  - We write

$$\Gamma \models \psi$$

    if and only if every interpretation that makes all sentences in Γ true also makes ψ true
  - We said that Γ entails ψ

Semantic of Propositional Logic

- Satisfiability vs. Validity vs. Entailment
  - ψ is valid iff True $\models \psi$ (also written $\models \psi$)
  - ψ is unsatisfiable iff $\psi \models$ False
  - $\Gamma \models \psi$ iff $\Gamma \cup \{\neg\psi\}$ is unsatisfiable

Inference in Propositional Logic

- Forward Chaining
- Backward Chaining

Forward Chaining

- Given a set of rules, i.e. formulae of the form

$$p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q$$

  and a set of known facts, i.e., formulae of the form

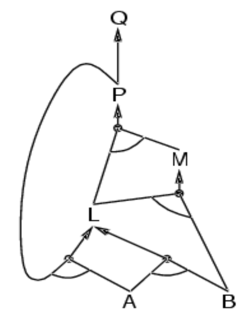
$$q,\ r, \ldots$$

- A new fact p is added
- Find all rules that have p as a premise
- If the other premises are already known to hold then
  - add the consequent to the set of know facts, and
  - trigger further inferences

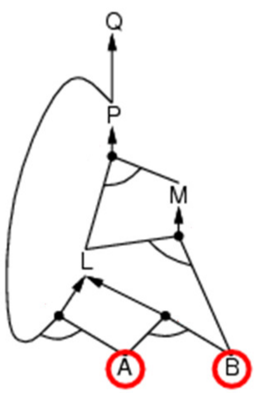
Forward Chaining

- Example

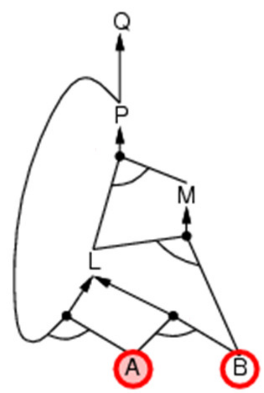
$$P \Rightarrow Q$$
$$L \wedge M \Rightarrow P$$
$$B \wedge L \Rightarrow M$$
$$A \wedge P \Rightarrow L$$
$$A \wedge B \Rightarrow L$$
$$A$$
$$B$$



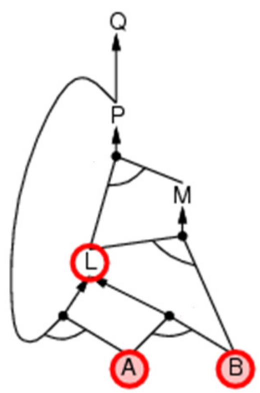
Forward Chaining



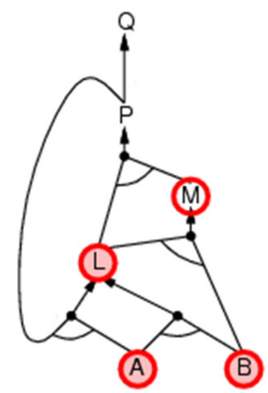
Forward Chaining



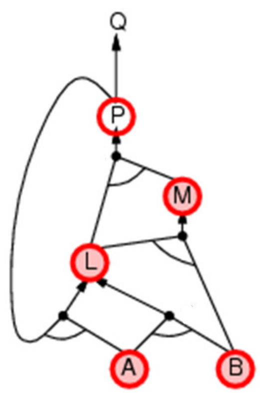
Forward Chaining



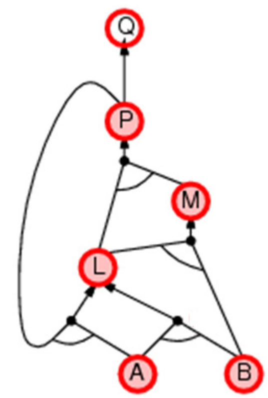
Forward Chaining



Forward Chaining



Forward Chaining



Forward Chaining

- Soundness
  - Yes
- Completeness
  - Yes

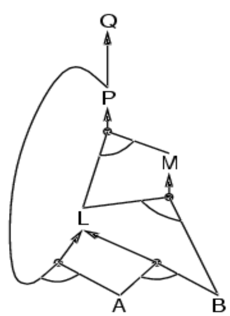
Backward Chaining

- Given a set of rules, and a set of known facts
- We ask whether a fact *P* is a consequence of the set of rules and the set of known facts
- The procedure check whether *P* is in the set of known facts
- Otherwise find all rules that have *P* as a consequent
  - If the premise is a conjunction, then process the conjunction conjunct by conjunct

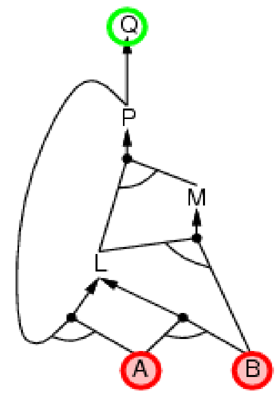
Backward Chaining

- Example

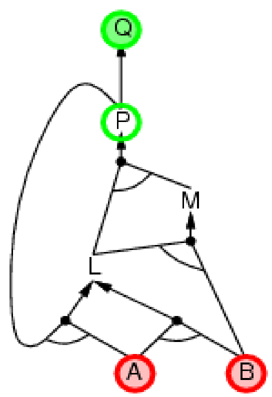
$$
\begin{aligned}
P &\Rightarrow Q \\
L \wedge M &\Rightarrow P \\
B \wedge L &\Rightarrow M \\
A \wedge P &\Rightarrow L \\
A \wedge B &\Rightarrow L \\
A & \\
B &
\end{aligned}
$$



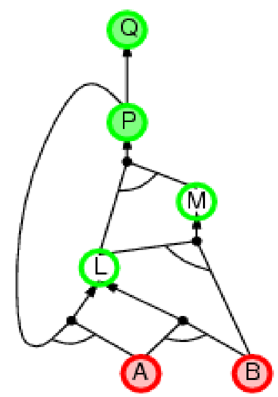
Backward Chaining



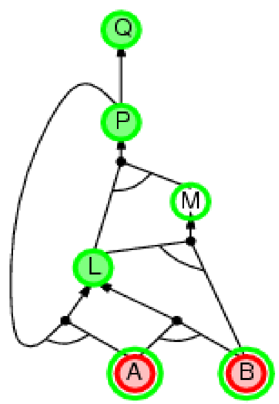
Backward Chaining



Backward Chaining



Backward Chaining



Backward Chaining

- Soundness
  - Yes
- Completeness
  - Yes

Transformation rules

$$(\alpha \wedge \beta) \equiv (\beta \wedge \alpha)$$
$$(\alpha \vee \beta) \equiv (\beta \vee \alpha)$$
giao hoán - swop

$$((\alpha \wedge \beta) \wedge \gamma) \equiv (\alpha \wedge (\beta \wedge \gamma))$$
$$((\alpha \vee \beta) \vee \gamma) \equiv (\alpha \vee (\beta \vee \gamma))$$
kết hợp - combination

$$\neg(\neg\alpha) \equiv \alpha$$
phủ định kép - double negation

$$(\alpha \Rightarrow \beta) \equiv (\neg\beta \Rightarrow \neg\alpha)$$
tương phản - contrast

$$(\alpha \Rightarrow \beta) \equiv (\neg\alpha \vee \beta)$$
$$(\alpha \Leftrightarrow \beta) \equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))$$

$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$$
$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$$
de Morgan

$$(\alpha \wedge (\beta \vee \gamma)) \equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))$$
$$(\alpha \vee (\beta \wedge \gamma)) \equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma))$$
phân phối -distribution

Transformation rules (con't)

Luật hấp thu:

- $(A \vee (A \wedge B) \equiv A$
- $(A \wedge (A \vee B)) \equiv A$

Các luật về 0, 1:

- $A \wedge 0 \Leftrightarrow 0$
- $A \vee 1 \Leftrightarrow 1$
- $\neg 1 \Leftrightarrow 0$
- $A \vee 0 \Leftrightarrow A$
- $A \wedge 1 \Leftrightarrow A$
- $\neg 0 \Leftrightarrow 1$

Luật bài trung:

- $\neg A \vee A \Leftrightarrow 1$

Luật mâu thuẫn:

- $\neg A \wedge A \Leftrightarrow 0$

Transform into CNF

$$
\begin{aligned}
(\alpha \wedge \beta) &\equiv (\beta \wedge \alpha)\\
(\alpha \vee \beta) &\equiv (\beta \vee \alpha)\\
((\alpha \wedge \beta) \wedge \gamma) &\equiv (\alpha \wedge (\beta \wedge \gamma))\\
((\alpha \vee \beta) \vee \gamma) &\equiv (\alpha \vee (\beta \vee \gamma))\\
\neg(\neg\alpha) &\equiv \alpha\\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha)\\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \vee \beta)\\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))\\
\neg(\alpha \wedge \beta) &\equiv (\neg\alpha \vee \neg\beta)\\
\neg(\alpha \vee \beta) &\equiv (\neg\alpha \wedge \neg\beta)\\
(\alpha \wedge (\beta \vee \gamma)) &\equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))\\
(\alpha \vee (\beta \wedge \gamma)) &\equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma))
\end{aligned}
$$

$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

1. Remove $\Leftrightarrow$, replace $\alpha \Leftrightarrow \beta$ by $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$.
   $(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

2. Remove $\Rightarrow$, replace $\alpha \Rightarrow \beta$ by $\neg\alpha \vee \beta$.
   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$

3. Move negation inward using the de Morgan's rule :
   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge ((\neg P_{1,2} \wedge \neg P_{2,1}) \vee B_{1,1})$

4. Applying the "and" distribution rule :
   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg P_{1,2} \vee B_{1,1}) \wedge (\neg P_{2,1} \vee B_{1,1})$

Example

$$
\begin{aligned}
(\alpha \wedge \beta) &\equiv (\beta \wedge \alpha)\\
(\alpha \vee \beta) &\equiv (\beta \vee \alpha)\\
((\alpha \wedge \beta) \wedge \gamma) &\equiv (\alpha \wedge (\beta \wedge \gamma))\\
((\alpha \vee \beta) \vee \gamma) &\equiv (\alpha \vee (\beta \vee \gamma))\\
\neg(\neg\alpha) &\equiv \alpha\\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha)\\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \vee \beta)\\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))\\
\neg(\alpha \wedge \beta) &\equiv (\neg\alpha \vee \neg\beta)\\
\neg(\alpha \vee \beta) &\equiv (\neg\alpha \wedge \neg\beta)\\
(\alpha \wedge (\beta \vee \gamma)) &\equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))\\
(\alpha \vee (\beta \wedge \gamma)) &\equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma))
\end{aligned}
$$

$(A \vee B) \rightarrow (C \rightarrow D)$

1. Remove $\Rightarrow$
   $\neg(A \vee B) \vee (\neg C \vee D)$
2. Move negation inward
   $(\neg A \wedge \neg B) \vee (\neg C \vee D)$
3. Distribution
   $(\neg A \vee \neg C \vee D) \wedge (\neg B \vee \neg C \vee D)$

Exercises

Transform the following expression into CNF.

1. $P \lor (\neg P \land Q \land R)$
2. $(\neg P \land Q) \lor (P \land \neg Q)$
3. $\neg(P \Rightarrow Q) \lor (P \lor Q)$
4. $(P \Rightarrow Q) \Rightarrow R$
5. $(P \Rightarrow (Q \Rightarrow R)) \Rightarrow ((P \land S) \Rightarrow R)$
6. $(P \land (Q \Rightarrow R)) \Rightarrow S$
7. $P \land Q \Rightarrow R \land S$
8. $((a \lor b) \land c) \rightarrow (c \land d)$

Priority: $\neg \; \land \; \lor \; \rightarrow \; \leftrightarrow$

Example

$$
\begin{aligned}
(\alpha \land \beta) &\equiv (\beta \land \alpha) \\
(\alpha \lor \beta) &\equiv (\beta \lor \alpha) \\
((\alpha \land \beta) \land \gamma) &\equiv (\alpha \land (\beta \land \gamma)) \\
((\alpha \lor \beta) \lor \gamma) &\equiv (\alpha \lor (\beta \lor \gamma)) \\
\neg(\neg\alpha) &\equiv \alpha \\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha) \\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \lor \beta) \\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)) \\
\neg(\alpha \land \beta) &\equiv (\neg\alpha \lor \neg\beta) \\
\neg(\alpha \lor \beta) &\equiv (\neg\alpha \land \neg\beta) \\
(\alpha \land (\beta \lor \gamma)) &\equiv ((\alpha \land \beta) \lor (\alpha \land \gamma)) \\
(\alpha \lor (\beta \land \gamma)) &\equiv ((\alpha \lor \beta) \land (\alpha \lor \gamma))
\end{aligned}
$$

1. $P \lor (\neg P \land Q \land R)$
2. $(\neg P \land Q) \lor (P \land \neg Q)$
3. $\neg(P \Rightarrow Q) \lor (P \lor Q)$
4. $(P \Rightarrow Q) \Rightarrow R$
5. $(P \Rightarrow (Q \Rightarrow R)) \Rightarrow ((P \land S) \Rightarrow R)$
6. $(P \land (Q \Rightarrow R)) \Rightarrow S$
7. $P \land Q \Rightarrow R \land S$

RESOLUTION

Convert all expressions in KB to standard form CNF

Apply consecutive rules in inferences (Resolution rule).

Begin: $(KB \wedge \neg\alpha)$

KB is a combination of expressions in the standard form CNF

Hence, $(KB \wedge \neg\alpha)$ is also expression in the standard form CNF!

The process of applying the conjugation rules stops when:

Having a conflict

After done by the resolution, getting (infer) the empty expression (conflict)

$$\frac{p, \quad \neg p}{\{\}}$$

No new expressions are expressed anymore

???

CHỨNG MINH BẰNG HỢP GIẢI: VÍ DỤ (1)

Giả sử có tập giả thiết KB

$p \wedge q$

$p \rightarrow r$

$(q \wedge r) \rightarrow s$

Cần chứng minh định lý *s*

Bước 1. Chuyển đổi KB về dạng chuẩn CNF

$(p \rightarrow r)$ được chuyển thành $(\neg p \vee r)$

$((q \wedge r) \rightarrow s)$ được chuyển thành $(\neg q \vee \neg r \vee s)$

Bước 2. Phủ định biểu thức cần chứng minh

$\neg s$

Bước 3. Áp dụng liên tiếp luật hợp giải đối với $(KB \wedge \neg\alpha)$:

$\{p, \; q, \; \neg p \vee r, \; \neg q \vee \neg r \vee s, \; \neg s\}$

RESOLUTION PROOF IN EXAMPLE

At the beginning of the solution process, we have a set of propositions :

1) p
2) q
3) ¬p ∨ r
4) ¬q ∨ ¬r ∨ s
5) ¬s

Combining 1) and 3), we get
r

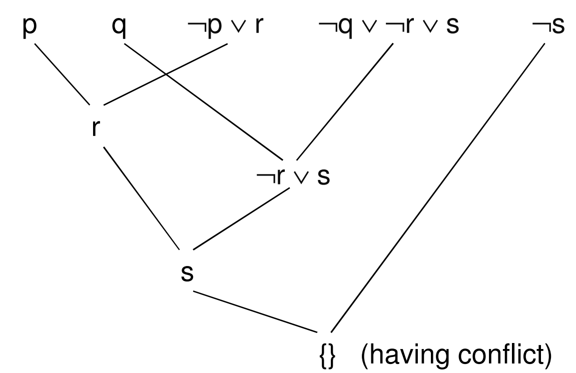
Combining 2) and 4), we get
7) ¬r ∨ s

Combining 6) and 7), we get
s

Combining 8) and 5), we get the conflict ({})

It mean that the original expression(s) is proven to be true

RESOLUTION PROOF IN EXAMPLE ...



Homework
due 9 Nov. 2022

- 1.Give an example of Forward Chaining and explain by steps
- 2.Give an example of Backward Chaining and explain by steps
- Remarks: a Student can draw a rule tree to represent these examples