PROPOSITIONAL LOGIC (Cont.)

Faculty of DS & AI
Autumn semester, 2026

Trong-Nghia Nguyen

Business AI Lab

Content

- Representing problems using logic
- Propositional logic
- **Deduction in Propositional Logic**

Deduction in Propositional Logic

Definition

- Deduction = deriving **conclusions** logically from known **premises**

Method	Approach	Characteristics
Truth Table	Exhaustive	Simple but exponential complexity
Inference Rules	Symbolic reasoning	More efficient and structured
Resolution (Robinson)	Contradiction-based	Basis for theorem proving

Deduction in Propositional Logic

Preparation clause

- Recall (from last week): converting any formula to CNF involves 4 steps: (eliminating $\Leftrightarrow$, eliminating $\Rightarrow$, pushing $\neg$ inwards, and distributing $\vee$ over $\wedge$).
- CNF is the **AND** of clauses;
- A clause is the **OR** of literals.

Clause types

- Unit clause : 1 literal
  - C1: e $\rightarrow$ "e is true" (fact)
  - C2: $\neg$w $\rightarrow$ "w is false" (negated goal)

- Binary clause : 2 literals
  - C3: $\neg$e $\vee$ b $\rightarrow$ "if e then b" (e $\Rightarrow$ b)
  - C4: $\neg$v $\vee$ w $\rightarrow$ "if v then w" (v $\Rightarrow$ w)

- General clause: $\geq$ 3 literals
  - C5: $\neg$d $\vee$ $\neg$m $\vee$ $\neg$b $\vee$ w
  - C6: $\neg$u $\vee$ v $\vee$ $\neg$f

- Empty clause $\perp$: NO literals $\rightarrow$ a CONTRADICTION (goal of refutation proof)

Deduction in Propositional Logic

Resolution rule

- **Resolution** is an inference rule that combines two clauses containing complementary literals to produce a new clause (resolvent)

Resolution Rule (General Form):

Clause 1: $A_1 \vee \dots \vee A_n \vee P$

Clause 2: $B_1 \vee \dots \vee B_m \vee \neg P$

Resolvent: $A_1 \vee \dots \vee A_n \vee B_1 \vee \dots \vee B_m$

where P and ¬P are complementary literals

Deduction in Propositional Logic

Prove using Robinson's method

Proof by CONTRADICTION

To prove $KB \vdash c$:

- assume the conclusion is FALSE → add ¬c to KB,
- show that $(p_1 \wedge p_2 \wedge \ldots \wedge p_n \wedge \neg c)$ leads to a contradiction.

$KB \vdash c \quad \equiv \quad KB \wedge \neg c \rightarrow \bot$ (empty clause)

$KB \wedge \neg c \xrightarrow{\text{resolution}} \ldots \longrightarrow \{ u , \neg u \} \longrightarrow \bot$

Robinson procedure

Begin

1. Add ¬c to the premise set G.
2. Convert all expressions in G to CNF. (4 steps — last week)
3. Split each conjunction into separate clauses.
4. Repeat:
   - 4.1 Choose two clauses A, B in G.
   - 4.2 If A, B are resolvable:
     - compute Res(A, B);
     - if it is a NEW clause, add it to G.

   Until the empty clause ⊥ is derived, OR no new clause can be generated.

End

Deduction in Propositional Logic

Prove using Robinson's method

Note when select clause: Unit Preference

- Always prioritize choosing **unit clause** (clause with 1 literal)
- Resolution with unit clause creates shorter clause
- Shorter clause → easier to find contradictions

Deduction in Propositional Logic

Prove using Robinson's method

Exercise 1

Given that the following premises are true:

1. $v \Rightarrow w$
2. $(d \wedge m \wedge b) \Rightarrow w$
3. $u \Rightarrow (v \vee \neg f)$
4. $(b \wedge c \wedge a) \Rightarrow v$
5. $(u \wedge e) \Rightarrow (\neg m \vee a)$
6. $e \Rightarrow (f \wedge m)$
7. $(e \wedge f) \Rightarrow u$
8. $((c \vee e) \Rightarrow b) \wedge e$
9. $b \Rightarrow (d \Rightarrow a)$

Prove that w and b are also true by Robinson's solution method (arbitrary flexible selection mechanism)

Deduction in Propositional Logic

Prove using Robinson's method

Exercise 1 Prove that **w** and **b** are true from the given **9 premises**.

A. Part 1: Proving b = TRUE

- **Step 0: Add ¬b to the premise set**
  - Assume b is false, add ¬b to the premise set G.

- **Step 1: Normalize to Conjunctive Normal Form (CNF)**

Original premises:

1. $v \Rightarrow w$
2. $(d \wedge m \wedge b) \Rightarrow w$
3. $u \Rightarrow (v \vee \neg f)$
4. $(b \wedge c \wedge a) \Rightarrow v$
5. $(u \wedge e) \Rightarrow (\neg m \vee a)$
6. $e \Rightarrow (f \wedge m)$
7. $(e \wedge f) \Rightarrow u$
8. $((c \vee e) \Rightarrow b) \wedge e$
9. $b \Rightarrow (d \Rightarrow a)$



CNF forms:

1. $\neg v \vee w$
2. $\neg d \vee \neg m \vee \neg b \vee w$
3. $\neg u \vee v \vee \neg f$
4. $\neg b \vee \neg c \vee \neg a \vee v$
5. $\neg u \vee \neg e \vee \neg m \vee a$
6. $(\neg e \vee f) \wedge (\neg e \vee m)$
7. $\neg e \vee \neg f \vee u$
8. $(\neg c \vee b) \wedge (\neg e \vee b) \wedge e$
9. $\neg b \vee \neg d \vee a$

Deduction in Propositional Logic

Prove using Robinson's method

Exercise 1 Prove that **w** and **b** are true from the given **9 premises**.

CNF forms:

1. ¬v ∨ w
2. ¬d ∨ ¬m ∨ ¬b ∨ w
3. ¬u ∨ v ∨ ¬f
4. ¬b ∨ ¬c ∨ ¬a ∨ v
5. ¬u ∨ ¬e ∨ ¬m ∨ a
6. (¬e ∨ f) ∧ (¬e ∨ m)
7. ¬e ∨ ¬f ∨ u
8. (¬c ∨ b) ∧ (¬e ∨ b) ∧ e
9. ¬b ∨ ¬d ∨ a



Step 2: Split into sub-clauses
Clause set G:

- C1: ¬v ∨ w
- C2: ¬d ∨ ¬m ∨ ¬b ∨ w
- C3: ¬u ∨ v ∨ ¬f
- C4: ¬b ∨ ¬c ∨ ¬a ∨ v
- C5: ¬u ∨ ¬e ∨ ¬m ∨ a
- C6a: ¬e ∨ f
- C6b: ¬e ∨ m
- C7: ¬e ∨ ¬f ∨ u
- C8a: ¬c ∨ b
- C8b: ¬e ∨ b
- C8c: e (fact from premise 8)
- C9: ¬b ∨ ¬d ∨ a
- C10: ¬b (negated conclusion - proof by contradiction)

Deduction in Propositional Logic

Prove using Robinson's method

Exercise 1 Prove that **w** and **b** are true from the given **9 premises**.

Step 2: Split into sub-clauses
Clause set G:
- C1: ¬v ∨ w
- C2: ¬d ∨ ¬m ∨ ¬b ∨ w
- C3: ¬u ∨ v ∨ ¬f
- C4: ¬b ∨ ¬c ∨ ¬a ∨ v
- C5: ¬u ∨ ¬e ∨ ¬m ∨ a
- C6a: ¬e ∨ f
- C6b: ¬e ∨ m
- C7: ¬e ∨ ¬f ∨ u
- C8a: ¬c ∨ b
- C8b: ¬e ∨ b
- C8c: e (fact from premise 8)
- C9: ¬b ∨ ¬d ∨ a
- C10: ¬b (negated conclusion - proof by contradiction)



Step 3: Resolution process
- Iteration 1: Select clauses C8c and C8b (flexible selection mechanism)
  - C8c: e
  - C8b: ¬e ∨ b
  - Res(C8c, C8b) = C11: b ✓ (new clause generated)
- Iteration 2: Select clauses C11 and C10
  - C11: b
  - C10: ¬b
  - Res(C11, C10) = ⊥(empty clause - CONTRADICTION!)

Conduct similar steps part 2: w

Deduction in Propositional Logic

Horn form

- An expression is in the Horn normal form if:
  - It is a conjunction (i.e., an AND combination) of clauses
  - Each clause is a disjunction (i.e., an OR combination) of literals and has at most 1 positive literal
  - Example: $(p \vee \neg q) \wedge (\neg p \vee \neg r \vee s)$
- Not all propositional expression can be converted to the Horn normal form!
- Representation of the set of premises KB in Horn normal form
  - **Rules**
    - $(\neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_n \vee q)$
    - Equivalent to the rule: $(p_1 \wedge p_2 \wedge \ldots \wedge p_n \rightarrow q)$
  - **Facts**
    - p, q
  - **Integrity constraints**
    - $(\neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_n)$
    - Equivalent to the rule: $(p_1 \wedge p_2 \wedge \ldots \wedge p_n \rightarrow \mathit{false})$

Deduction in Propositional Logic

Generalized Modus Ponens rule

$$\frac{(p_1 \wedge p_2 \wedge \ldots \wedge p_n \rightarrow q),\ p_1,\ p_2,\ \ldots,\ p_n}{q}$$

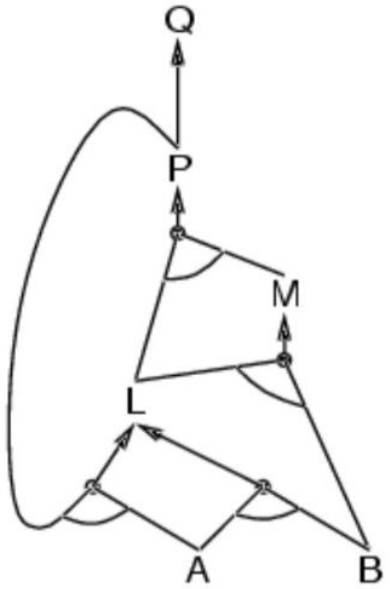
- The Modus Ponens rule is *sound* and *complete*, provided that propositional symbols and the set of premises KB are in Horn normal form
- The Modus Ponens rule can be used by both of the 2 reasoning approaches: *Forward reasoning* and *Backward reasoning*

Deduction in Propositional Logic

Forward reasoning (chaining)

- Given a set of premises (knowledge base) *KB*, it requires to prove the expression *Q*
- Idea: Repeat the following 2 steps until inferring the expression
  - Apply a rule whose condition (IF) part is satisfied in *KB*
  - Add the applied rule's conclusion (THEN) part to *KB*

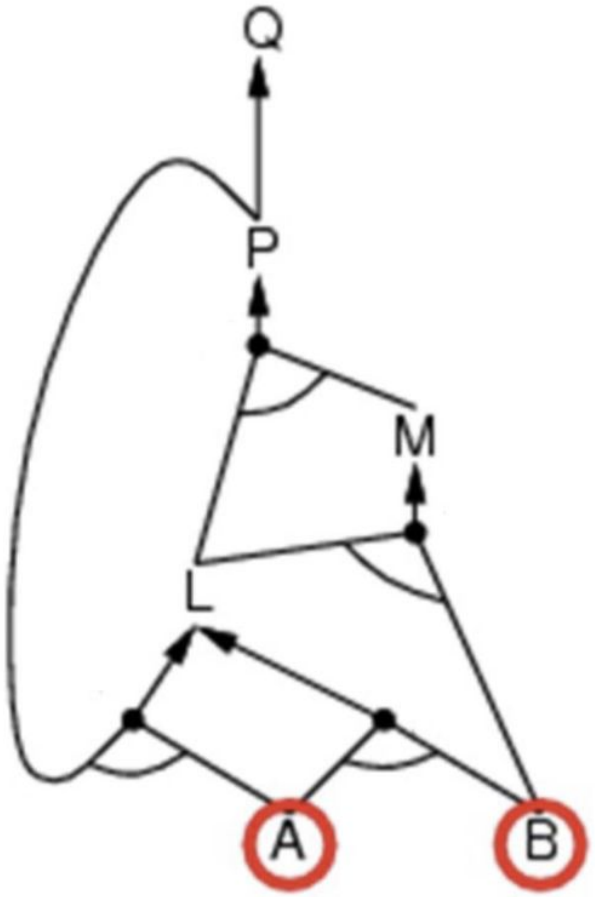
$$P \Rightarrow Q$$
$$L \wedge M \Rightarrow P$$
$$B \wedge L \Rightarrow M$$
$$A \wedge P \Rightarrow L$$
$$A \wedge B \Rightarrow L$$
$$A$$
$$B$$



Deduction in Propositional Logic

Forward reasoning (chaining)

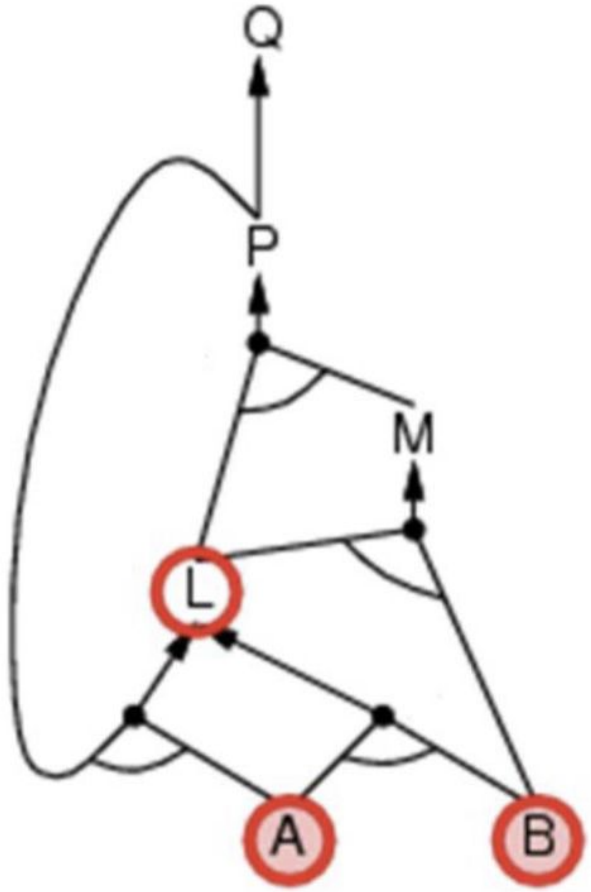
$$P \Rightarrow Q$$
$$L \wedge M \Rightarrow P$$
$$B \wedge L \Rightarrow M$$
$$A \wedge P \Rightarrow L$$
$$A \wedge B \Rightarrow L$$
$$A$$
$$B$$



Deduction in Propositional Logic

Forward reasoning (chaining)

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



Deduction in Propositional Logic

Forward reasoning (chaining)

$P \Rightarrow Q$

$L \wedge M \Rightarrow P$

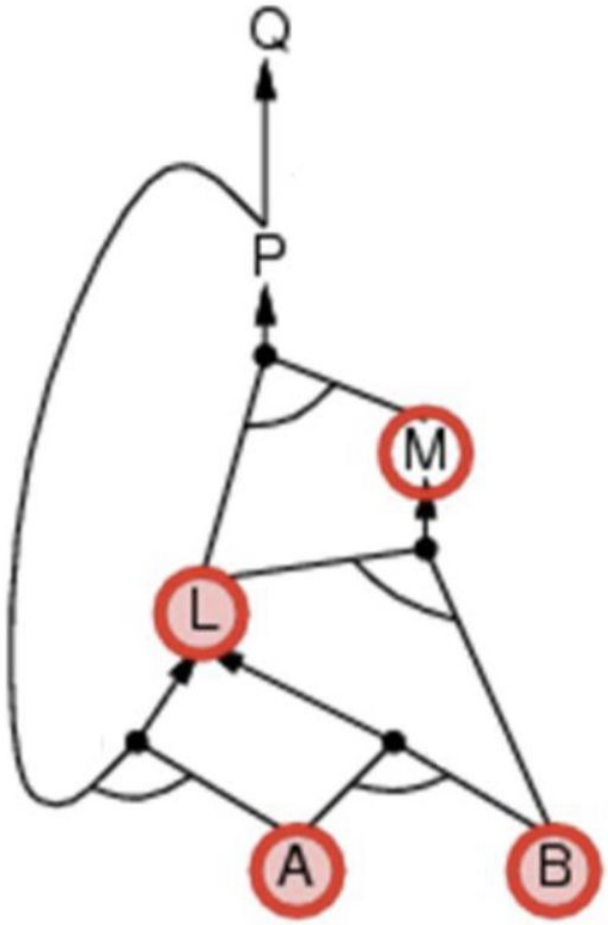
$B \wedge L \Rightarrow M$

$A \wedge P \Rightarrow L$

$A \wedge B \Rightarrow L$

A

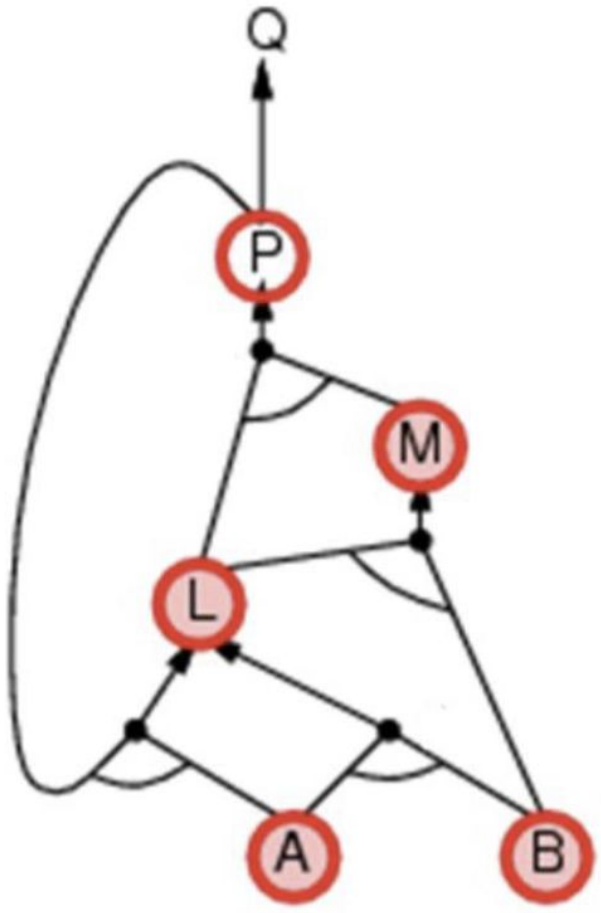
B



Deduction in Propositional Logic

Forward reasoning (chaining)

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



Deduction in Propositional Logic

Forward reasoning (chaining)

$P \Rightarrow Q$

$L \wedge M \Rightarrow P$

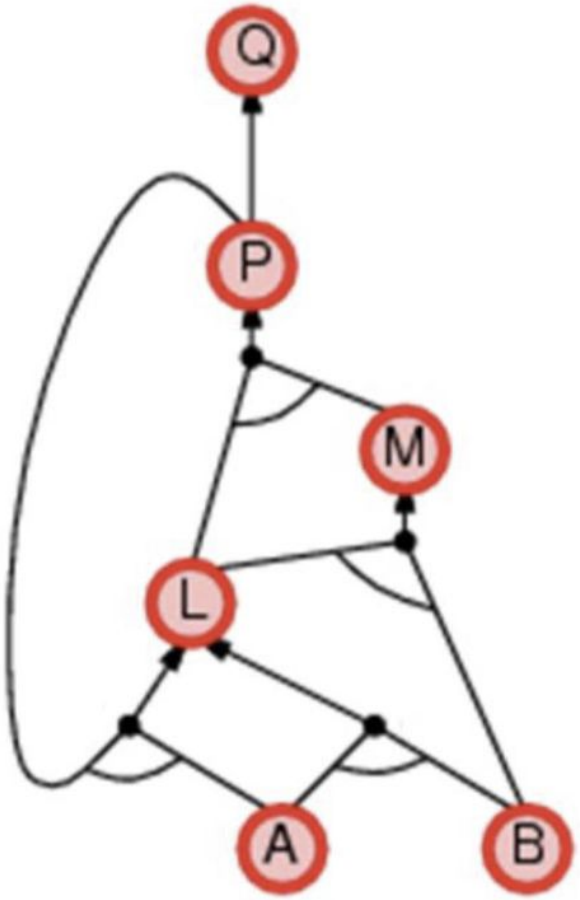
$B \wedge L \Rightarrow M$

$A \wedge P \Rightarrow L$

$A \wedge B \Rightarrow L$

A

B



Deduction in Propositional Logic

Backward reasoning (chaining)

- Idea: The reasoning process starts from the conclusion *Q*
- To prove *Q* by the set of premises (i.e., knowledge base) *KB*
  - Check if Q has been proven by *KB*,
  - If not yet, continue proving all the conditions of a rule (in *KB*) whose conclusion is *Q*
- Avoid loops
  - Check if the new expressions have been included in the list of expressions to prove? – If yes, then do not include them again!
- Avoid proving again to an expression
  - Has previously been proven true
  - Has previously been proven unsatisfiable (i.e., false) in *KB*

Deduction in Propositional Logic

Backward reasoning (chaining)

$P \Rightarrow Q$

$L \wedge M \Rightarrow P$

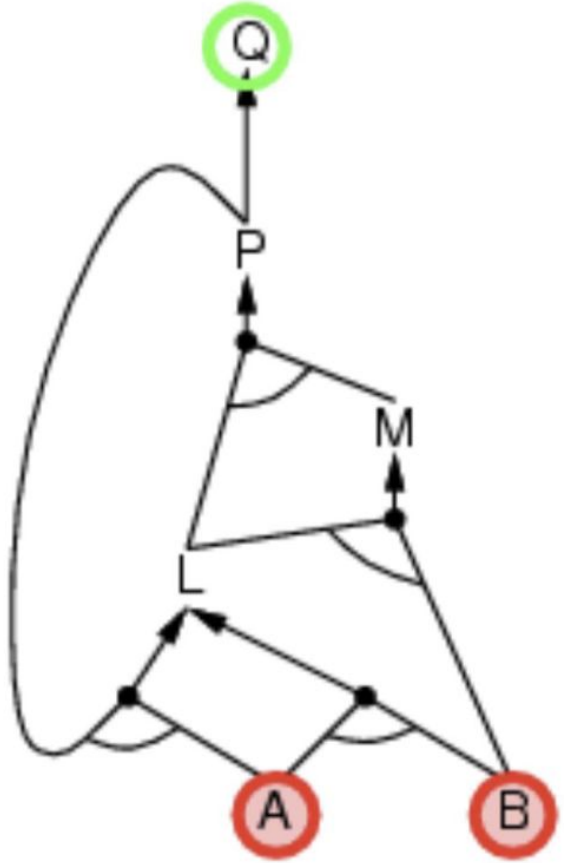
$B \wedge L \Rightarrow M$

$A \wedge P \Rightarrow L$

$A \wedge B \Rightarrow L$

A

B



Deduction in Propositional Logic

Backward reasoning (chaining)

$P \Rightarrow Q$

$L \wedge M \Rightarrow P$

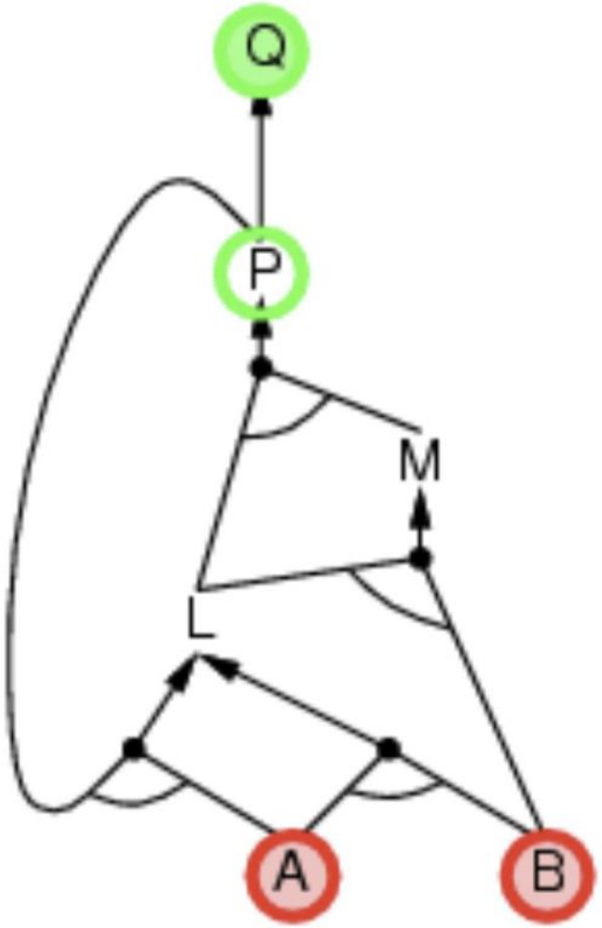
$B \wedge L \Rightarrow M$

$A \wedge P \Rightarrow L$

$A \wedge B \Rightarrow L$

A

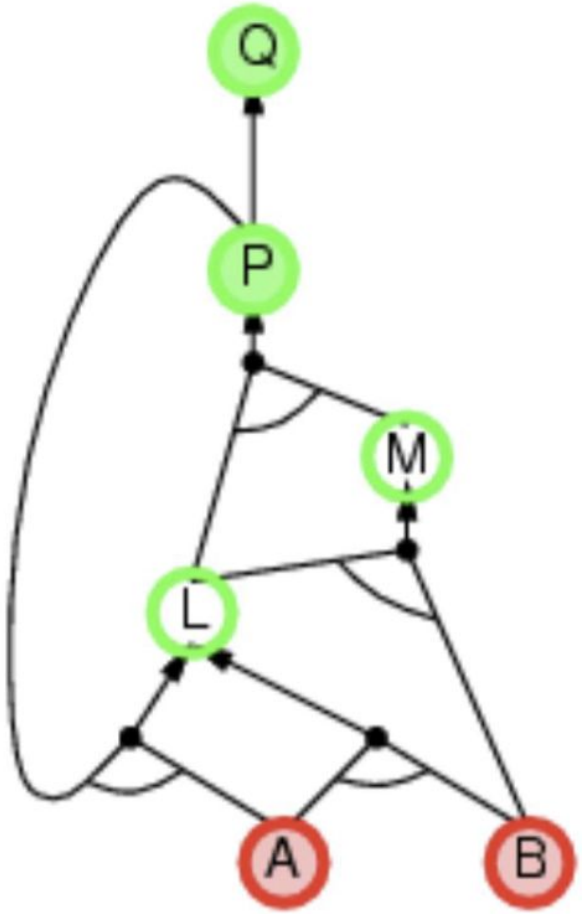
B



Deduction in Propositional Logic

Backward reasoning (chaining)

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



Deduction in Propositional Logic

Backward reasoning (chaining)

$P \Rightarrow Q$

$L \wedge M \Rightarrow P$

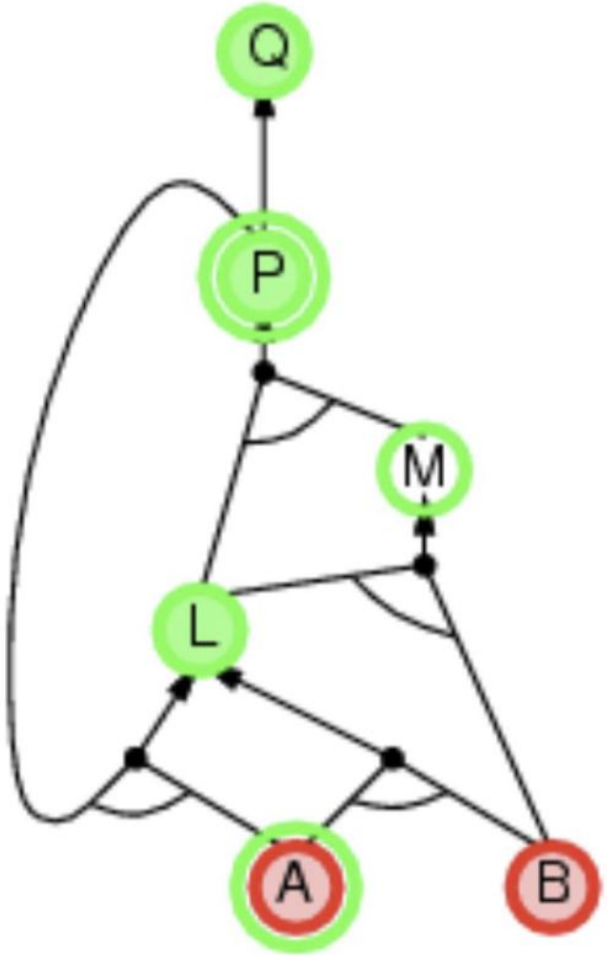
$B \wedge L \Rightarrow M$

$A \wedge P \Rightarrow L$

$A \wedge B \Rightarrow L$

A

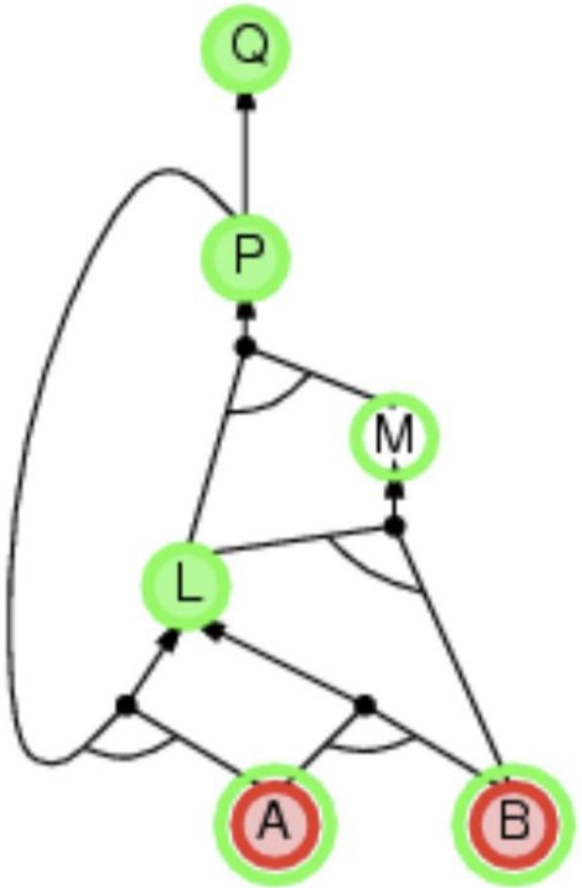
B



Deduction in Propositional Logic

Backward reasoning (chaining)

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



Deduction in Propositional Logic

Backward reasoning (chaining)

- Forward reasoning is a data-driven process
  - Example: object recognition, decision making
- Forward reasoning may perform many redundant inference steps – irrelevant (unnecessary) to the proving goal
- Backward reasoning is a goal-driven process, suitable for problem solving

Deduction in Propositional Logic

Exercises 2

Symbols:

Symbol	Meaning	Symbol	Meaning
T	Tet is coming	V	Visit new, beautiful places
B	There is a bonus	M	Visit My Dinh
F	Have fun	G	Visit Grandma
P	Go out / play		

Rules (Horn):

R1: T ∧ B ⇒ F
R2: F ⇒ P
R3: P ⇒ V
R4: V ⇒ M
R5: M ⇒ G

Facts: T, B. **Query:** prove **G** ("we will visit Grandma").

Tasks:

1. Solve by **forward chaining**: list the firing order of the rules and the propositions derived.
2. Solve by **backward chaining**: draw the goal–subgoal decomposition tree.
3. **(Extension)** Add rule **R6: B ⇒ Shopping**. What does forward chaining additionally derive? Does backward chaining (goal G) ever use R6? State your observation.

Deduction in Propositional Logic

Exercises 3

Symbols: Rain = it is raining · Wet = the road is wet · Slippery = the road is slippery · Driving = you are driving · Danger = danger.

Rules (Horn):

```
R1: Rain                  ⇒ Wet
R2: Wet                   ⇒ Slippery
R3: Slippery ∧ Driving ⇒ Danger
```

Facts: Rain, Driving. **Query:** prove **Danger**.

Tasks: solve by **forward chaining** and by **backward chaining**; point out where the AND-rule (R3) is used.

Thank you!