



AUTOMATA, FORMAL LANGUAGES AND LOGIC

Dr Pooja Agarwal

Professor,
Department of Computer Science & Engineering

AUTOMATA, FORMAL LANGUAGES AND LOGIC

MODULE 5

Propositional Logic

Dr Pooja Agarwal

Department of Computer Science & Engineering

Outline

- Inference and proofs
 - Theorem proving
 - Logical equivalence
 - Validity
 - Satisfiability
 - Inference and Proofs
 - Modus Ponens
 - And-Elimination

Theorem proving

- Applying **rules of inference** directly to the sentences in our knowledge base, to construct a proof of the desired sentence without consulting models

AUTOMATA FORMAL LANGUAGES AND LOGIC

Inference and Proofs



Logical equivalence

Validity

Satisfiability

Logical equivalence

- Two sentences α and β are logically equivalent if they are true in the same set of models.
- We write this as $\alpha \equiv \beta$.
- For example, $P \wedge Q$ and $Q \wedge P$ are logically equivalent

- Two sentences are **logically equivalent** iff true in same models:
 $\alpha \equiv \beta$ iff $\alpha \models \beta$ and $\beta \models \alpha$

$$(\alpha \wedge \beta) \equiv (\beta \wedge \alpha) \quad \text{commutativity of } \wedge$$

$$(\alpha \vee \beta) \equiv (\beta \vee \alpha) \quad \text{commutativity of } \vee$$

$$((\alpha \wedge \beta) \wedge \gamma) \equiv (\alpha \wedge (\beta \wedge \gamma)) \quad \text{associativity of } \wedge$$

$$((\alpha \vee \beta) \vee \gamma) \equiv (\alpha \vee (\beta \vee \gamma)) \quad \text{associativity of } \vee$$

$$\neg(\neg\alpha) \equiv \alpha \quad \text{double-negation elimination}$$

$$(\alpha \Rightarrow \beta) \equiv (\neg\beta \Rightarrow \neg\alpha) \quad \text{contraposition}$$

$$(\alpha \Rightarrow \beta) \equiv (\neg\alpha \vee \beta) \quad \text{implication elimination}$$

$$(\alpha \Leftrightarrow \beta) \equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination}$$

$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta) \quad \text{de Morgan}$$

$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta) \quad \text{de Morgan}$$

$$(\alpha \wedge (\beta \vee \gamma)) \equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) \quad \text{distributivity of } \wedge \text{ over } \vee$$

$$(\alpha \vee (\beta \wedge \gamma)) \equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) \quad \text{distributivity of } \vee \text{ over } \wedge$$

Validity

- A sentence is valid if it is true in all models.
- For example, the sentence $\mathbf{P \vee \neg P}$ is valid
- Validity is connected to inference via **deduction theorem**
- For any sentences α and β , $\alpha \models \beta$ if and only if the sentence $(\alpha \Rightarrow \beta)$ is valid.

Satisfiability

- A sentence is satisfiable if it is true in, or satisfied by, some model
- Satisfiability can be checked by enumerating the possible models *until one is found* that satisfies the sentence.

AUTOMATA FORMAL LANGUAGES AND LOGIC

Inference and Proofs



Examples

- Decide whether each of the following sentences is valid, unsatisfiable, or satisfiable
 - Smoke \Rightarrow Smoke *Valid*
 - Smoke \Rightarrow Fire *Satisfiable*
 - $((\text{Smoke} \wedge \text{Heat}) \Rightarrow \text{Fire}) \Leftrightarrow ((\text{Smoke} \Rightarrow \text{Fire}) \wedge (\text{Heat} \Rightarrow \text{Fire}))$ *Valid*

Inference Rules

- Modus Ponens
- And-Elimination

Inference Rules

Modus Ponens

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

- Whenever any sentences of the form $\alpha \Rightarrow \beta$ and α are given, then the sentence β can be inferred
- For example, if $(WumpusAhead \wedge WumpusAlive) \Rightarrow Shoot$ and $(WumpusAhead \wedge WumpusAlive)$ are given, then $Shoot$ can be inferred
- If today is Sunday (p) then it is a holiday(q).
- Today is Sunday(p). *It infers, It is a holiday*

Inference Rules

And-Elimination

- from a conjunction, **any of the** conjuncts can be inferred

$$\frac{\alpha \wedge \beta}{\alpha}$$

- For example, from *(WumpusAhead \wedge WumpusAlive)*,
WumpusAlive can be inferred.

- Example

- We have $KB = R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5$.
We want to prove $\neg P_{1,2}$

$$R_1: \neg P_{1,1}$$

$$R_2: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$R_3: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$$

$$R_4: \neg B_{1,1}$$

$$R_5: B_{2,1}$$

$$R_6: (B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}) \text{ by} \\ \text{bicond. elim } R_2$$

$$R_7: ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}) \text{ by And-Elimination to } R_6$$

$$R_8: (\neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})) \text{ by contrapositives}$$

$$R_9: \neg(P_{1,2} \vee P_{2,1}) \text{ by Modus Ponens with } R_8 \text{ and } R_4$$

$$R_{10}: \neg P_{1,2} \wedge \neg P_{2,1} \text{ by De Morgan's rule}$$

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



THANK YOU

Dr Pooja Agarwal

Department of Computer Science and
Engineering

poojaagarwal@pes.edu