Fundamentals of Artificial Intelligence
Exercise 4: Logical Agents

Florian Lercher

Technical University of Munich

Tweedback

tweedback.de/zv8y

Friday 1st December, 2023

Problem 4.1: Model, satisfaction relation, and entailment

Which of the following statements are correct? Prove correctness by reasoning about the models satisfying each sentence.

1. $\mathit{False} \models \mathit{True}$
2. $\mathit{True} \models \mathit{False}$
3. $(A \land B) \models (A \Leftrightarrow B)$
4. $(A \Leftrightarrow B) \models (A \lor B)$
5. $(A \Leftrightarrow B) \models (\neg A \lor B)$

Reminder: Entailment

Entailment

Entailment is the relationship between two sentences where the truth of one sentence requires the truth of the other sentence, which is written as

$$\alpha \models \beta$$

if α entails β. Formally, entailment is defined as

$$\alpha \models \beta \text{ if and only if } M(\alpha) \subseteq M(\beta).$$

For instance, the sentence $x = 0$ entails $xy = 0$.

Models:

A	B	$A \vee B$
T	T	T
T	F	T
F	T	T
F	F	F

$\mathcal{M}(A \vee B) = \{(T,T), (T,F), (F,T)\}$

$A \models B$ iff $A \Rightarrow B$ is valid

$\longleftrightarrow$

$\models$ vs. $\Rightarrow$:

$\mathbb{N} \xrightarrow{\text{is talked about by}}$ Theorems

0, 1, 2, 3

$\mathbb{N} \longrightarrow$ Algorithms $\longrightarrow$ Complexity

Proposition $\longrightarrow$ Sentences $\longrightarrow$ Meta-Logic

$A \Rightarrow B$ $\qquad A \wedge A \models A$

Problem 4.1: Model, satisfaction relation, and entailment

1. *False* $\models$ *True*

$\iff M(False) \subseteq M(True)$

$\iff \emptyset \subseteq$ All models

$\iff$ Correct

"Ex falso quodlibet"

$False \models \alpha$

Problem 4.1: Model, satisfaction relation, and entailment

2. $True \models False$

$\hookrightarrow \mathcal{M}(True) \subseteq \mathcal{M}(False)$

$\hookrightarrow$ All Models $\subseteq \emptyset$

$\neq$

$\emptyset$

$\hookrightarrow$ Incorrect

$True \not\models False$

$(T), (F)$

$\{()\} \neq \emptyset$

Problem 4.1: Model, satisfaction relation, and entailment

3. $(A \land B) \models (A \Leftrightarrow B)$

$\Leftrightarrow \mathcal{M}(A \land B) \subseteq \mathcal{M}(A \Leftrightarrow B)$

$\Leftrightarrow \{(T,T)\} \subseteq \{(T,T),(F,F)\}$

$\Leftrightarrow$ Correct

A	B	$A \land B$	$A \Leftrightarrow B$
T	T	T $\leftarrow$	T $\leftarrow$
T	F	F	F
F	T	F	F
F	F	F	T $\leftarrow$

Problem 4.1: Model, satisfaction relation, and entailment

4. $(A \Leftrightarrow B) \not\models (A \vee B)$

$\Longleftrightarrow \mathcal{M}(A \Leftrightarrow B) \subseteq \mathcal{M}(A \vee B)$

$\Longleftrightarrow \{(T,T), (F,F)\} \subseteq \{(T,T), (T,F), (F,T)\}$

$\times$

$\Longleftrightarrow$ Incorrect

Problem 4.1: Model, satisfaction relation, and entailment

5. $(A \Leftrightarrow B) \models (\neg A \vee B)$

$\Leftrightarrow \mathcal{M}(A \Leftrightarrow B) \subseteq \mathcal{M}(\neg A \vee B)$

$\Leftrightarrow \{(T,T), (F,F)\} \subseteq \{(T,T), \cancel{(T,F)}, (F,T), (F,F)\}$

$\Leftrightarrow$ Correct

Problem 4.2: Validity, satisfiability, and unsatisfiability

Problem 4.2.1: Prove the following two metatheorems:

1. Sentence α is valid if and only if $\alpha \equiv \mathit{True}$,
2. Sentence α is unsatisfiable if and only if $\alpha \equiv \mathit{False}$.

$A \equiv B$ iff $A \models B$ and $B \models A$

Reminder: Validity and satisfiability

Validity

A sentence is valid if it is true in **all** models (e.g. $P \vee \neg P$). Valid sentences are also known as **tautologies**.

Satisfiability

A sentence is satisfiable if it is true in **some** model. E.g. the expression $P_1 \wedge P_2$ is satisfiable for $P_1 = P_2 = true$, whereas $P_1 \wedge \neg P_1$ is not satisfiable.

Valid $\Rightarrow$ Satisfiable

- The problem of determining the satisfiability of sentences is also called **SAT** problem, which is NP-complete.
- Validity and satisfiability are connected: α is valid if $\neg\alpha$ is unsatisfiable.

Problem 4.2: Validity, satisfiability, and unsatisfiability

1. Sentence α is valid if and only if $\underline{\alpha \equiv \textit{True}}$

$\alpha \equiv True$

$\iff \alpha \models True$ and $True \models \alpha$

$\iff \mathcal{M}(\alpha) \subseteq \mathcal{M}(True)$ and $\mathcal{M}(True) \subseteq \mathcal{M}(\alpha)$

$\iff \mathcal{M}(\alpha) = \mathcal{M}(True) = $ All models

$\iff \alpha$ is valid

Problem 4.2: Validity, satisfiability, and unsatisfiability

2. Sentence α is unsatisfiable if and only if $\alpha \equiv$ *False*

α is unsatisfiable

$\Leftrightarrow \mathcal{M}(\alpha) = \emptyset = \mathcal{M}(False)$

$\Leftrightarrow \mathcal{M}(\alpha) \subseteq \mathcal{M}(False)$ and $\mathcal{M}(False) \subseteq \mathcal{M}(\alpha)$

$\Leftrightarrow \alpha \models False$ and $False \models \alpha$

$\Leftrightarrow \alpha \equiv False$

Problem 4.2: Validity, satisfiability, and unsatisfiability

Problem 4.2.2: Show whether each of the following sentences is valid, satisfiable, or unsatisfiable. To this end, use the two metatheorems above, the standard logical equivalences from the lecture, and the following four logical equivalences:

$$\alpha \lor \neg\alpha \equiv \mathit{True} \qquad \alpha \lor \alpha \equiv \alpha$$

$$\alpha \land \neg\alpha \equiv \mathit{False} \qquad \alpha \land \alpha \equiv \alpha$$

1. $\mathit{Smoke} \Rightarrow \mathit{Smoke}$
2. $(\mathit{Smoke} \Rightarrow \mathit{Fire}) \Rightarrow (\neg\mathit{Smoke} \Rightarrow \neg\mathit{Fire})$
3. $\mathit{Smoke} \lor \mathit{Fire} \lor \neg\mathit{Fire}$
4. $(\mathit{Fire} \Rightarrow \mathit{Smoke}) \land \mathit{Fire} \land \neg\mathit{Smoke}$

Reminder: Logical equivalences

Standard logical equivalences

$$
\begin{aligned}
(\alpha \land \beta) &\equiv (\beta \land \alpha) && \text{commutativity of } \land \\
(\alpha \lor \beta) &\equiv (\beta \lor \alpha) && \text{commutativity of } \lor \\
((\alpha \land \beta) \land \gamma) &\equiv (\alpha \land (\beta \land \gamma)) && \text{associativity of } \land \\
((\alpha \lor \beta) \lor \gamma) &\equiv (\alpha \lor (\beta \lor \gamma)) && \text{associativity of } \lor \\
\neg(\neg\alpha) &\equiv \alpha && \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha) && \text{contraposition} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \lor \beta) && \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)) && \text{biconditional elimination} \\
\neg(\alpha \land \beta) &\equiv (\neg\alpha \lor \neg\beta) && \text{De Morgan} \\
\neg(\alpha \lor \beta) &\equiv (\neg\alpha \land \neg\beta) && \text{De Morgan} \quad a \cdot (b+c) = ab + ac \\
(\alpha \land (\beta \lor \gamma)) &\equiv ((\alpha \land \beta) \lor (\alpha \land \gamma)) && \text{distributivity of } \land \text{ over } \lor \\
(\alpha \lor (\beta \land \gamma)) &\equiv ((\alpha \lor \beta) \land (\alpha \lor \gamma)) && \text{distributivity of } \lor \text{ over } \land
\end{aligned}
$$

Problem 4.2: Validity, satisfiability, and unsatisfiability

1. $Smoke \Rightarrow Smoke$

$\equiv \neg Smoke \lor Smoke$

$\equiv True$

Valid, satisfiable

Problem 4.2: Validity, satisfiability, and unsatisfiability

2. $(Smoke \Rightarrow Fire) \Rightarrow (\neg Smoke \Rightarrow \neg Fire)$

$\equiv \neg(Smoke \Rightarrow Fire) \lor (\neg Smoke \Rightarrow \neg Fire)$

$\equiv \neg(\neg Smoke \lor Fire) \lor (\cancel{\neg}\neg Smoke \lor \neg Fire)$

$\equiv (Smoke \land \neg Fire) \lor (Smoke \lor \neg Fire)$

$\equiv (Smoke \lor \neg Fire \lor Smoke) \land (\neg Fire \lor Smoke \lor \neg Fire)$

$\equiv (Smoke \lor \neg Fire) \land (\neg Fire \lor Smoke)$

$\equiv Smoke \lor \neg Fire$

Smoke = T   Fire = F ✓ is a model

Smoke = F   Fire = T ✗ is not a model

Satisfiable, but not valid

Problem 4.2: Validity, satisfiability, and unsatisfiability

3. $Smoke \lor \underline{Fire \lor \neg Fire}$

$\equiv Smoke \lor True$

$\equiv True$

$\leadsto$ valid, satisfiable

Problem 4.2: Validity, satisfiability, and unsatisfiability

4. $(Fire \Rightarrow Smoke) \land Fire \land \neg Smoke$

$\equiv (\neg Fire \lor Smoke) \land Fire \land \neg Smoke$

$\equiv \underbrace{(\neg Fire \land Fire \land \neg Smoke)}_{} \lor (Smoke \land Fire \land \neg Smoke)$

$\equiv \quad False \quad \lor \quad False$

$\equiv \quad False$

$\leadsto$ unsatisfiable

Problem 4.3: Knights and Knaves

Suppose we are on an island with two types of inhabitants: “knights” who always tell the truth, and “knaves” who always lie.

> *According to this problem, three of the inhabitants – A, B and C – were standing together in the garden. A stranger passed by and asked A, “Are you a knight or a knave?”. A answered, but rather indistinctly, so the stranger could not make out what he said. The stranger then asked B, “What did A say?”. B replied, “A said that he is a knave”. At this point the third man, C, said “Don’t believe B; he’s lying!”. The question is, what are B and C?*

Model this logic puzzle by introducing three atomic propositions A, B, and C with intended interpretation that A, B, and C are knights.

A is true iff A is a knight

Problem 4.3: Knights and Knaves

Problem 4.3.1: How can you formalize the sentence “A says that B is a knight”?

Case 1: A is a knight (A is true)

→ B is a knight (B is true)

Case 2: A is a knave (A is false)

→ B is a knave (B is false)

A is true iff B is true

$A \Leftrightarrow B$

Problem 4.3: Knights and Knaves

Problem 4.3.2: Assume that *Remark* represents what a person says and that we can represent it using propositional logic. Additionally, assume that P could either be A, B, or C. From the previous problem, can you generalize the method to model the sentence "person P says (or replies) *Remark*"?

Case 1: P is a knight
↪ Remark is true

Case 2: P is a knave
↪ Remark is false

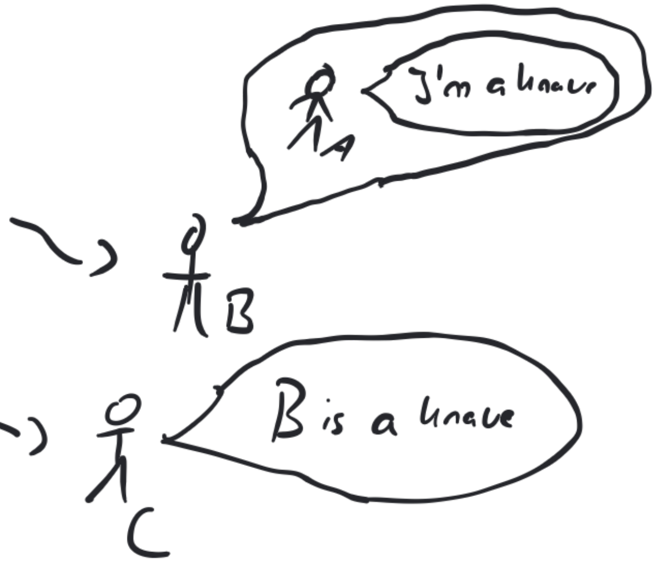
⇝ $P \iff Remark$

Problem 4.3: Knights and Knaves

Problem 4.3.3: Model the following facts which are taken from the puzzle:

$\neg A$

1. B replies, "A said that he is a knave."
2. C says, "Don't believe B; he's lying!"



1. $B \Leftrightarrow (A \Leftrightarrow \neg A)$

2. $C \Leftrightarrow \neg B$

Problem 4.3: Knights and Knaves

Problem 4.3.4: By using the following logical equivalences

$$(X \Leftrightarrow \neg X) \equiv \mathit{False}$$
$$(X \Leftrightarrow \mathit{False}) \equiv \neg X$$

and the following deduction (inference) rule

$$\frac{P \Leftrightarrow Q \qquad Q}{P}$$

deduce what B and C are.

Problem 4.3: Knights and Knaves

Problem 4.3.4:

1. $B \Leftrightarrow \underline{(A \Leftrightarrow \neg A)}$

$\equiv B \Leftrightarrow \text{False}$

$\equiv \underline{\neg B}$

$\leadsto B$ is a knave

2. $$\frac{C \Leftrightarrow \neg B \qquad \neg B}{C}$$

$\leadsto C$ is a knight

Problem 4.4: Superman does not exist

↳ or "The Problem of Evil" ⇝ Theodicy

If Superman were able and willing to prevent evil, he would do so. If Superman were unable to prevent evil, he would be impotent; if he were unwilling to prevent evil, he would be malevolent. Superman does not prevent evil. If Superman exists, he is neither impotent nor malevolent. Therefore, Superman does not exist.

Assume that we use the following propositions and their meaning:

A : Superman is able to prevent evil.
W : Superman is willing to prevent evil.
I : Superman is impotent.
M : Superman is malevolent.
P : Superman prevents evil.
E : Superman exists.

Problem 4.4: Superman does not exist

Problem 4.4.1: Formalize the facts from the text using the propositions defined above.

Problem 4.4: Superman does not exist

1. If Superman were able and willing to prevent evil, he would do so.

   $(A \wedge W) \Rightarrow P$ $\qquad$ $\underbrace{(A \wedge W)}_{\text{not a sentence}} \models \underbrace{P}_{\text{sentences}}$

2. If Superman were unable to prevent evil, he would be impotent.

   $\neg A \Rightarrow I$

3. If he were unwilling to prevent evil, he would be malevolent.

   $\neg W \Rightarrow M$

4. Superman does not prevent evil.

   $\neg P$

5. If Superman exists, he is neither impotent nor malevolent.

   $E \Rightarrow (\neg I \wedge \neg M)$

Problem 4.4: Superman does not exist

Problem 4.4.2: Assume we want to prove that "Superman does not exist" using the resolution approach for propositional logic. Identify which sentences belong to the knowledge base KB, and which sentence we want to deduce. How do we need to process these sentences before applying the resolution principle?

Reminder: Resolution algorithm

Inference procedures based on resolution use the principle of **proof by contradiction**:

$\equiv$ False

To show that $KB \models \alpha$, we show that $KB \wedge \neg\alpha$ is unsatisfiable.

Basic procedure

1. $KB \wedge \neg\alpha$ is converted into CNF
2. The resolution rule is applied to the resulting clauses:
   each pair that contains complementary literals is resolved to produce a new clause, which is added to the others (if not already present)
3. The process continues until
   - there are no new clauses to be added $\Rightarrow KB \not\models \alpha$;
   - two clauses resolve to yield the *empty* clause $\Rightarrow KB \models \alpha$.

Reminder: Resolution rule

Full resolution rule

$$\frac{l_1 \vee \ldots \vee l_k, \quad m_1 \vee \ldots \vee m_n}{l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee \ldots \vee l_k \vee m_1 \vee \ldots \vee m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n},$$

where l_i and m_j are complementary literals.

eliminate complementary literals and keep all others

$$\frac{A \vee B \qquad \neg B \vee C}{A \vee C}$$

Reminder: Conjunctive Normal Form

- The resolution rule only applies to disjunction of literals, which are also called **clauses**.
- Fortunately, every sentence of propositional logic can be reformulated as a conjunction of clauses, which is also referred to as **conjunctive normal form (CNF)**

Conjunctive Normal Form

A sentence with literals x_{ij} of the form $\bigwedge_i \bigvee_j (\neg) x_{ij}$ is in conjunctive normal form.

Examples:

- $(A \vee B \vee C) \wedge (\neg A \vee B \vee C)$ yes
- $A \wedge B \wedge C \vee (\neg A \wedge B \vee C)$ no
- $A \wedge B \wedge C \wedge (\neg A \vee B \vee C)$ yes

Problem 4.4: Superman does not exist

Problem 4.4.2: Conversion to CNF

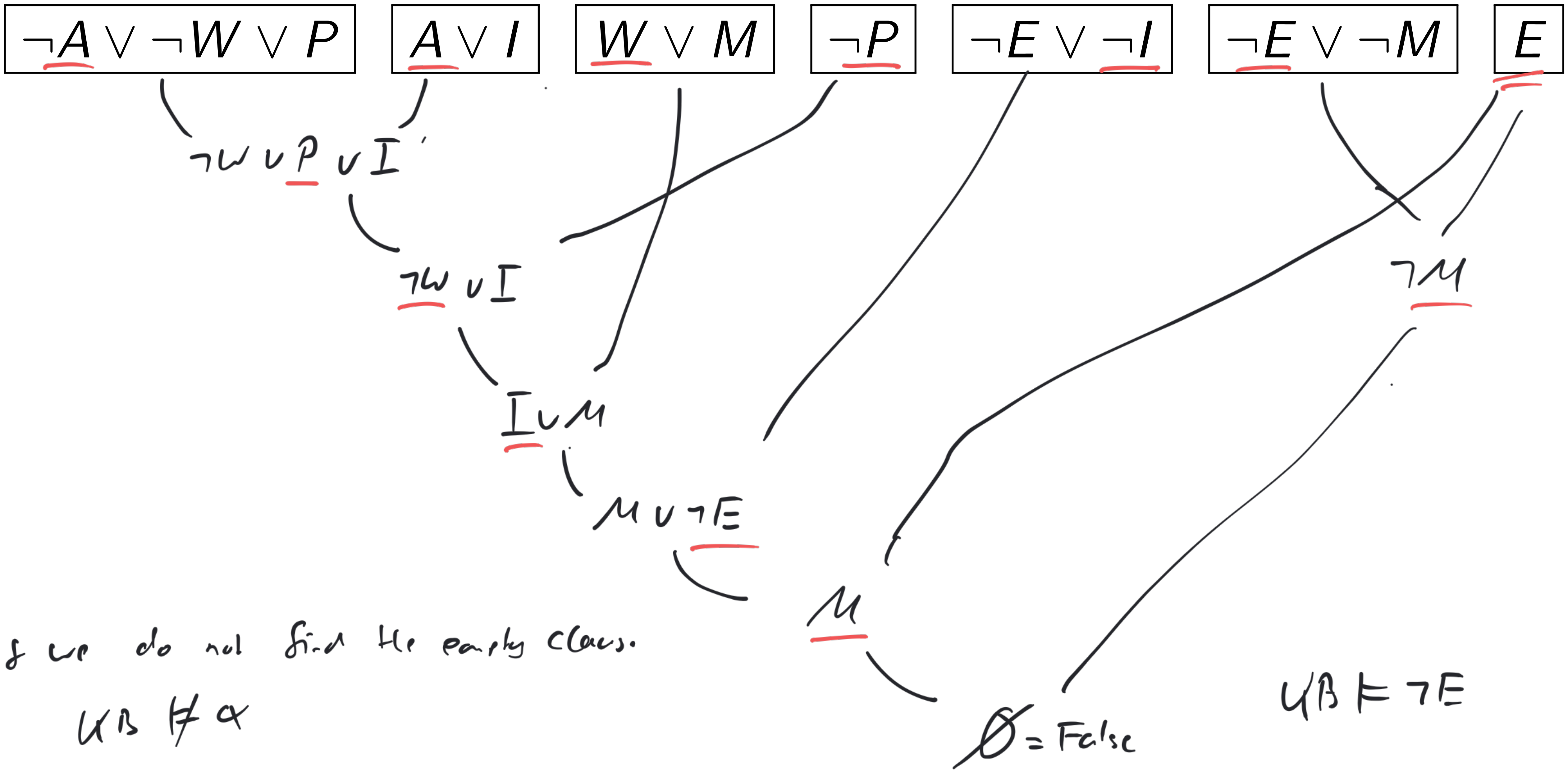
Knowledge base:

1. $A \wedge W \Rightarrow P \equiv \neg(A \wedge W) \vee P \equiv \boxed{\neg A \vee \neg W \vee P}$
2. $\neg A \Rightarrow I \equiv \neg\neg A \vee I \equiv \boxed{A \vee I}$
3. $\neg W \Rightarrow M \equiv \boxed{W \vee M}$
4. $\boxed{\neg P}$
5. $E \Rightarrow \neg I \wedge \neg M \equiv \neg E \vee (\neg I \wedge \neg M) \equiv \boxed{(\neg E \vee \neg I)} \wedge \boxed{(\neg E \vee \neg M)}$

Goal: $\neg E \leadsto \neg\neg E \equiv \boxed{E}$

Problem 4.4: Superman does not exist

Problem 4.4.3: Prove diagramatically with the resolution approach that "Superman does not exist."



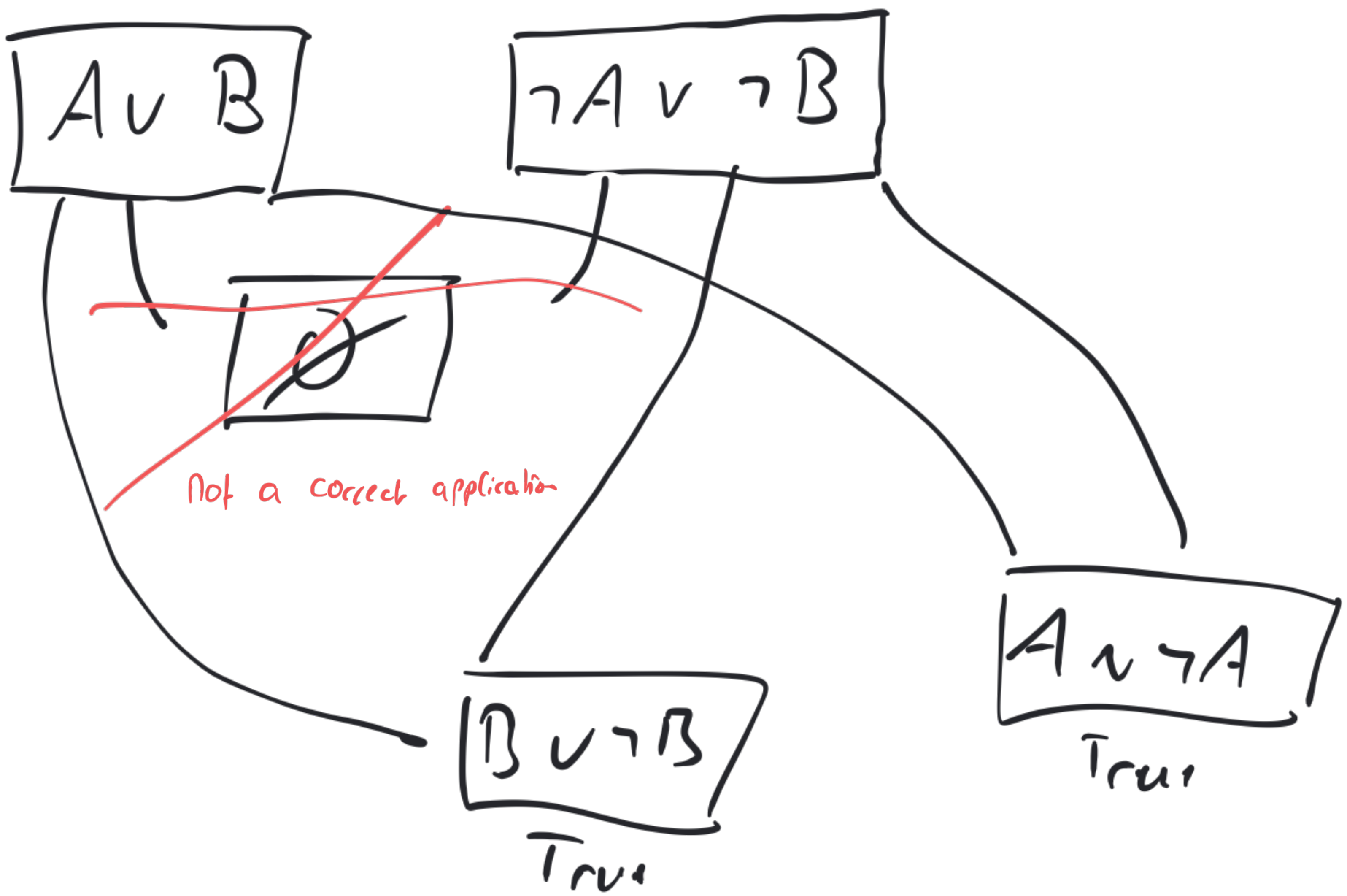
If we do not find the empty clause.

$KB \nvDash \alpha$

$KB \vDash \neg E$

Caveat for the resolution rule

tubk.de/zv8y



$A = T$ $B = F$

$\rightsquigarrow$ Always make sure to eliminate only one literal!

Problem 4.5: Completeness and soundness

Recall the definition of *completeness* and *soundness*.

> **Completeness:** An inference algorithm is complete if and only if for every entailed sentence $KB \models \alpha$, the inference algorithm will always be able to derive it.
>
> **Soundness:** An inference algorithm is sound if and only if for every sentence it derives, it is guaranteed that the sentence is entailed $KB \models \alpha$.

Completeness: "Every entailed sentence is derivable"

Soundness: "Every derived sentence is also entailed"

Problem 4.5: Completeness and soundness

tubh.de/zv8y

Problem 4.5.1: Suppose that we have an inference algorithm which will *always* be able to derive a given sentence (regardless whether it is entailed or not). Would this inference algorithm be complete? Sound?

Complete, but unsound

Problem 4.5: Completeness and soundness

twbk.de/zv8y

Problem 4.5.2: Suppose now that we have an inference algorithm which will *never* be able to derive a given sentence (regardless whether it is entailed or not). Would this inference algorithm be complete? Sound?

Sound, but incomplete