Logic 2: Modal Logic

Lecture 4

Wolfgang Schwarz

January 28, 2022

University of Edinburgh

Review

Review

The possible-worlds analysis of $\Box$ and $\Diamond$:

$\Box A$ says that A is true at all worlds.
$\Diamond A$ says that A is true at some world.

Review

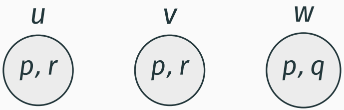
Under a given interpretation of p, $\Diamond p$ is true at a world in a scenario iff ...

...p is true at some world in the scenario.

We can represent scenarios and interpretations by models.

A **(basic) model** for $\mathfrak{L}_M$ is a pair of

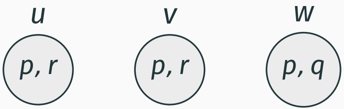
- a non-empty set W, and
- an interpretation function V that assigns to every sentence letter a subset of W.



A sentence is **valid** iff it is true at all worlds in all models.

Review

A model (M):



- Is $\Diamond\Box\neg(p \to r)$ true at u in M?
- Is $\Diamond\Box\neg(p \to r)$ true at all worlds in M?
- Is $\Diamond\Box\neg(p \to r)$ true at all worlds in all models?

The tree method

The tree method

The tree method (a.k.a. the method of analytic tableau) is a method for determining whether a sentence is valid or invalid.

Suppose we want to find out whether $p \to (q \to (r \vee p))$ is valid (in classical propositional logic).

We start by negating the target sentence:

$$1. \quad \neg(p \to (q \to (r \vee p))) \qquad \text{(Ass.)}$$

Our aim is to derive a contradiction from this assumption.

We have an assumption of the form $\neg(A \to B)$.

$A \to B$ is false only if A is true and B is false.

So our assumption entails p and $\neg(q \to (r \vee p))$.

We add these consequence below line 1.

The tree method

Target sentence: $p \to (q \to (r \vee p))$

1.	$\neg(p \to (q \to (r \vee p)))$	(Ass.)
2.	p	(1)
3.	$\neg(q \to (r \vee p))$	(1)

Line 3 also has the form $\neg(A \to B)$.

$A \to B$ is false only if A is true and B is false.

The tree method

Target sentence: $p \to (q \to (r \lor p))$

1.	$\neg(p \to (q \to (r \lor p)))$	(Ass.)
2.	p	(1)
3.	$\neg(q \to (r \lor p))$	(1)
4.	q	(3)
5.	$\neg(r \lor p)$	(3)

Line 5 has the form $\neg(A \lor B)$.

$A \lor B$ is false only if A and B are both false.

The tree method

Target sentence: $p \to (q \to (r \vee p))$

1.	$\neg(p \to (q \to (r \vee p)))$	(Ass.)
2.	p	(1)
3.	$\neg(q \to (r \vee p))$	(1)
4.	q	(3)
5.	$\neg(r \vee p)$	(3)
6.	$\neg r$	(5)
7.	$\neg p$	(5)
	x	

Assumption 1 has led to a contradiction: 2 and 7.

The tree method

Tree construction rules

1. To show that a sentence is valid, start the tree with its negation.
2. Then expand all nodes on the tree until no more nodes can be expanded.
3. To expand a non-negated node, you consider what the truth of the relevant sentence entails for the truth-values of its immediate parts. You then add these consequences to the tree.
4. To expand a negated node $\neg A$, you consider what the falsity of A entails for the truth-values of A's immediate parts. You then add these consequences to the tree.

The tree method

Target sentence: $(p \rightarrow q) \rightarrow ((q \rightarrow r) \rightarrow (p \rightarrow r))$

1. $\neg((p \rightarrow q) \rightarrow ((q \rightarrow r) \rightarrow (p \rightarrow r)))$ (Ass.)

The tree method

Target sentence: $(p \to q) \to ((q \to r) \to (p \to r))$

1.	$\neg((p \to q) \to ((q \to r) \to (p \to r)))$	(Ass.)
2.	$p \to q$	(1)
3.	$\neg((q \to r) \to (p \to r))$	(1)

The tree method

Target sentence: $(p \to q) \to ((q \to r) \to (p \to r))$

1.	$\neg((p \to q) \to ((q \to r) \to (p \to r)))$	(Ass.)
2.	$p \to q$	(1)
3.	$\neg((q \to r) \to (p \to r))$	(1)
4.	$q \to r$	(3)
5.	$\neg(p \to r)$	(3)

The tree method

Target sentence: $(p \to q) \to ((q \to r) \to (p \to r))$

1.	$\neg((p \to q) \to ((q \to r) \to (p \to r)))$	(Ass.)
2.	$p \to q$	(1)
3.	$\neg((q \to r) \to (p \to r))$	(1)
4.	$q \to r$	(3)
5.	$\neg(p \to r)$	(3)
6.	p	(5)
7.	$\neg r$	(5)

How do we expand assumptions 2 and 4?

The tree method

$$2. \qquad p \rightarrow q \qquad (1)$$

What can we infer from the truth of $p \rightarrow q$ about the truth-value of the immediate parts, p and q?

Either p is false or q is true.

We need to consider both possibilities.

1.	$\neg((p \to q) \to ((q \to r) \to (p \to r)))$	(Ass.)
2.	$p \to q$	(1)
3.	$\neg(q \to r) \to (p \to r))$	(1)
4.	$q \to r$	(3)
5.	$\neg(p \to r)$	(3)
6.	p	(5)
7.	$\neg r$	(5)

8. $\neg p$ (2)	9. q (2)
X	

The tree method

1. $\neg((p \to q) \to ((q \to r) \to (p \to r)))$ (Ass.)
2. $p \to q$ (1)
3. $\neg(q \to r) \to (p \to r))$ (1)
4. $q \to r$ (3)
5. $\neg(p \to r)$ (3)
6. p (5)
7. $\neg r$ (5)

8. $\neg p$ 9. q (2)

x

10. $\neg q$ 11. r (4)

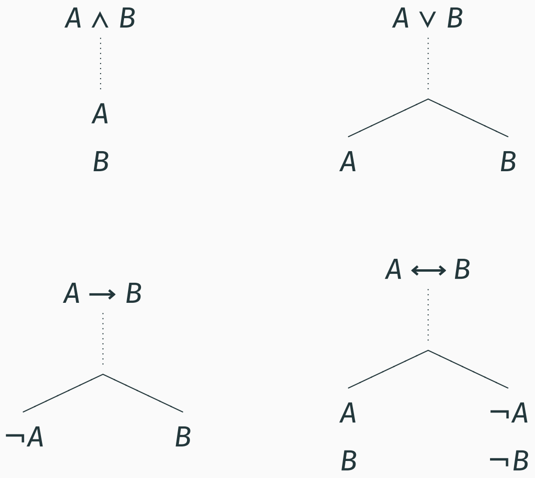
x x

Tree construction rules

1. To show that a sentence is valid, start the tree with its negation.
2. Then expand all nodes on the tree until no more nodes can be expanded.
3. If a branch of a tree contains a sentence A and its negation $\neg A$, the branch is closed with an 'x'.
4. When a node is expanded, the new nodes can be added to all open branches below the expanded node.

To keep your trees small, always expand non-branching nodes first.

The tree method



The tree method

$\neg(A \wedge B)$

$\neg A \qquad \neg B$

$\neg(A \vee B)$

$\neg A$

$\neg B$

$\neg(A \rightarrow B)$

A

$\neg B$

$\neg(A \leftrightarrow B)$

$A \qquad \neg A$

$\neg B \qquad B$

$\neg\neg A$

A

Modal tree rules

Modal tree rules

Let's show that $\Box p \rightarrow p$ is valid.

Modal sentences are true or false relative to a world.

So our starting assumption is that $\Box p \rightarrow p$ is false at some world w.

1.	$\neg(\Box p \rightarrow p)$	(w)	(Ass.)

Our goal is to derive a contradiction from this assumption.

Modal tree rules

Target: $\Box p \rightarrow p$

1.	$\neg(\Box p \rightarrow p)$	(w)	(Ass.)

If $\Box p \rightarrow p$ is false at w, then $\Box p$ is true at w and p is false at w.

Modal tree rules

Target: $\Box p \rightarrow p$

1.	$\neg(\Box p \rightarrow p)$	(w)	(Ass.)
2.	$\Box p$	(w)	(1)
3.	$\neg p$	(w)	(1)

If $\Box p$ is true at w, then p is true at all worlds, including w.

Modal tree rules

Target: $\Box p \rightarrow p$

1.	$\neg(\Box p \rightarrow p)$	(w)	(Ass.)
2.	$\Box p$	(w)	(1)
3.	$\neg p$	(w)	(1)
4.	p	(w)	(2)
	x		

p cannot be both true and false at w.

Modal tree rules

Target sentence: $p \rightarrow \Box\Diamond p$

1.	$\neg(p \rightarrow \Box\Diamond p)$	(w)	(Ass.)

Modal tree rules

Target sentence: $p \rightarrow \Box\Diamond p$

1.	$\neg(p \rightarrow \Box\Diamond p)$	(w)	(Ass.)
2.	p	(w)	(1)
3.	$\neg\Box\Diamond p$	(w)	(1)

Modal tree rules

Target sentence: $p \to \Box\Diamond p$

1.	$\neg(p \to \Box\Diamond p)$	(w)	(Ass.)
2.	p	(w)	(1)
3.	$\neg\Box\Diamond p$	(w)	(1)
4.	$\neg\Diamond p$	(v)	(3)

Modal tree rules

Target sentence: $p \rightarrow \Box\Diamond p$

1.	$\neg(p \rightarrow \Box\Diamond p)$	(w)	(Ass.)
2.	p	(w)	(1)
3.	$\neg\Box\Diamond p$	(w)	(1)
4.	$\neg\Diamond p$	(v)	(3)
5.	$\neg p$	(v)	(4)

Modal tree rules

Target sentence: $p \rightarrow \Box\Diamond p$

1.	$\neg(p \rightarrow \Box\Diamond p)$	(w)	(Ass.)
2.	p	(w)	(1)
3.	$\neg\Box\Diamond p$	(w)	(1)
4.	$\neg\Diamond p$	(v)	(3)
5.	$\neg p$	(v)	(4)
6.	$\neg p$	(w)	(4)
	x		

Modal tree rules

$$\begin{array}{ll} \Box A & (\omega) \\ \vdots & \\ A & (\nu) \\ & \uparrow \\ & \text{old} \end{array} \qquad \begin{array}{ll} \Diamond A & (\omega) \\ \vdots & \\ A & (\nu) \\ & \uparrow \\ & \text{new} \end{array}$$

$$\begin{array}{ll} \neg\Box A & (\omega) \\ \vdots & \\ \neg A & (\nu) \\ & \uparrow \\ & \text{new} \end{array} \qquad \begin{array}{ll} \neg\Diamond A & (\omega) \\ \vdots & \\ \neg A & (\nu) \\ & \uparrow \\ & \text{old} \end{array}$$

Modal tree rules

Tree construction rules

1. To show that a sentence is valid, start the tree with the negation at world w.
2. Then expand all nodes on the tree until no more nodes can be expanded.
3. If a branch of a tree contains a sentence A and its negation $\neg A$ at the same world, the branch is closed with an 'x'.
4. Nodes of type $\Box A$ and $\neg\Diamond A$ can be expanded multiple times, for each world on any branch to which the node belongs.
5. When a node of a type other than $\Box A$ and $\neg\Diamond A$ is expanded, and the new nodes have been added to all open branches below the expanded node, then the node is never expanded again.