CS:4980 Topics in Computer Science II

Introduction to Automated Reasoning

Propositional Logic Basics

Cesare Tinelli

Spring 2024

The University of Iowa

Propositional Logic

- Syntax
- Semantics, Satisfiability, and Validity
- Proof by deduction

Automating Inference

Automated Reasoning tries to automated the process of *inference*:

deriving consequences of a given set of statements

In AR, both the given and the derived knowledge are expressed in a formal language

Automating Inference

Automated Reasoning tries to automated the process of *inference*:

deriving consequences of a given set of statements

In AR, both the given and the derived knowledge are expressed in a formal language

Formal Languages for Knowledge Representation

Unlike natural languages (such as English), formal languages allow us to represent knowledge in a precise, unambiguous way

Just as importantly, statements in a formal language are machine-processable

Formal Languages for Knowledge Representation

Unlike natural languages (such as English), formal languages allow us to represent knowledge in a precise, unambiguous way

Just as importantly, statements in a formal language are machine-processable

Formal Logics

Formal languages for knowledge representation and reasoning

Many (formal) logics have been developed and studied,
with various degrees of expressiveness and mechanizability

We will consider a couple in this course, starting with the most basic one: Propositional Logic (PL)

Formal Logics

Formal languages for knowledge representation and reasoning

Many (formal) logics have been developed and studied,
with various degrees of expressiveness and mechanizability

We will consider a couple in this course, starting with the most basic one: Propositional Logic (PL)

Formal Logics

Formal languages for knowledge representation and reasoning

Many (formal) logics have been developed and studied,
with various degrees of expressiveness and mechanizability

We will consider a couple in this course, starting with the most basic one: Propositional Logic (PL)

Defining features of formal logics

A *formal logic* is

- defined by its syntax and semantics
- equipped with one or more inference/proof systems

syntax:	a set of symbols and rules for combining them to form sentences (formulas) of the logic
semantics:	a systematic, math-based way to assign meaning to sentences
proof system:	a system of formal rules of inference

Defining features of formal logics

A *formal logic* is

- defined by its syntax and semantics
- equipped with one or more inference/proof systems

syntax: a set of symbols and rules for combining them to form sentences (formulas) of the logic

semantics: a systematic, math-based way to assign meaning to sentences

proof system: a system of formal rules of inference

Defining features of formal logics

A *formal logic* is

- defined by its syntax and semantics
- equipped with one or more inference/proof systems

syntax: a set of symbols and rules for combining them to form sentences (formulas) of the logic

semantics: a systematic, math-based way to assign meaning to sentences

proof system: a system of formal rules of inference

Defining features of formal logics

A *formal logic* is

- defined by its syntax and semantics
- equipped with one or more inference/proof systems

syntax: a set of symbols and rules for combining them to form sentences (formulas) of the logic

semantics: a systematic, math-based way to assign meaning to sentences

proof system: a system of formal rules of inference

Defining features of formal logics

A *formal logic* is

- defined by its syntax and semantics
- equipped with one or more inference/proof systems

syntax:	a set of symbols and rules for combining them to form sentences (formulas) of the logic
semantics:	a systematic, math-based way to assign meaning to sentences
proof system:	a system of formal rules of inference

Classical logics

Formalize natural language statements that can be either true or false (but not both)

Classical logics

Formalize natural language statements that can be either true or false (but not both)

Basic sentences are called *atomic*

Examples:

1. $0 < 1$
2. Iowa City is in Iowa
3. $1 + 1 = 10$

Classical logics

Formalize natural language statements that can be either true or false (but not both)

More complex sentences are built from simpler ones via a small number of constructs

Examples:

1. If Iowa City is in Iowa then University Height is Iowa
2. $1 + 1 = 10$ or $1 + 1 \neq 10$

Truth of atomic sentences

Each proposition formalizes a statement that is either true or false

The *truth value* (true or false) of an atomic proposition P depends on P's *interpretation*

Example What is the truth value of the equality $1 + 1 = 10$?

- it is false, if we interpret 1 and 10 as integers in decimal notation (and $+$ as addition)
- it is true, if we interpret 1 and 10 as integers in binary notation (and $+$ as addition)

Truth of atomic sentences

Each proposition formalizes a statement that is either true or false

The *truth value* (true or false) of an atomic proposition P depends on P's *interpretation*

Example What is the truth value of the equality $1 + 1 = 10$?

- it is false, if we interpret 1 and 10 as integers in decimal notation (and $+$ as addition)
- it is true, if we interpret 1 and 10 as integers in binary notation (and $+$ as addition)

Truth of atomic sentences

Each proposition formalizes a statement that is either true or false

The *truth value* (true or false) of an atomic proposition P depends on P's *interpretation*

Example What is the truth value of the equality $1 + 1 = 10$?

- it is false, if we interpret 1 and 10 as integers in decimal notation (and $+$ as addition)
- it is true, if we interpret 1 and 10 as integers in binary notation (and $+$ as addition)

Truth of atomic sentences

Each proposition formalizes a statement that is either true or false

The *truth value* (true or false) of an atomic proposition P depends on P's *interpretation*

Example What is the truth value of the equality $1 + 1 = 10$?

- it is false, if we interpret 1 and 10 as integers in decimal notation (and $+$ as addition)
- it is true, if we interpret 1 and 10 as integers in binary notation (and $+$ as addition)

Truth of atomic sentences

Each proposition formalizes a statement that is either true or false

The *truth value* (true or false) of an atomic proposition P depends on P's *interpretation*

Example What is the truth value of the equality $1 + 1 = 10$?

- it is false, if we interpret 1 and 10 as integers in decimal notation (and $+$ as addition)
- it is true, if we interpret 1 and 10 as integers in binary notation (and $+$ as addition)

Truth of complex sentences

Let α be a complex sentence built with a construct c from simpler sentences $\alpha_1, \ldots, \alpha_n$

The truth value of α is uniquely determined by

1. the meaning of c
2. the truth value of $\alpha_1, \ldots, \alpha_n$

More precisely, it is a function (determined by c) of the truth values of $\alpha_1, \ldots, \alpha_n$

Example

is true if at least one of $1 + 1 = 5$, $1 + 1 \neq 5$ is true

Truth of complex sentences

Let α be a complex sentence built with a construct c from simpler sentences $\alpha_1, \ldots, \alpha_n$

The truth value of α is uniquely determined by

1. the meaning of c
2. the truth value of $\alpha_1, \ldots, \alpha_n$

More precisely, it is a function (determined by c) of the truth values of $\alpha_1, \ldots, \alpha_n$

Example

is true if at least one of $1 + 1 = 5$, $1 + 1 \neq 5$ is true

Truth of complex sentences

Let α be a complex sentence built with a construct c from simpler sentences $\alpha_1, \ldots, \alpha_n$

The truth value of α is uniquely determined by

1. the meaning of c
2. the truth value of $\alpha_1, \ldots, \alpha_n$

More precisely, it is a function (determined by c) of the truth values of $\alpha_1, \ldots, \alpha_n$

Example

is true if at least one of $1 + 1 = 5$, $1 + 1 \neq 5$ is true

Truth of complex sentences

Let α be a complex sentence built with a construct c from simpler sentences $\alpha_1, \ldots, \alpha_n$

The truth value of α is uniquely determined by

1. the meaning of c
2. the truth value of $\alpha_1, \ldots, \alpha_n$

More precisely, it is a function (determined by c) of the truth values of $\alpha_1, \ldots, \alpha_n$

Example

$$1 + 1 = 5 \text{ or } 1 + 1 \neq 5$$

is true if at least one of $1 + 1 = 5$, $1 + 1 \neq 5$ is true

Truth of complex sentences

Let α be a complex sentence built with a construct c from simpler sentences $\alpha_1, \ldots, \alpha_n$

The truth value of α is uniquely determined by

1. the meaning of c
2. the truth value of $\alpha_1, \ldots, \alpha_n$

More precisely, it is a function (determined by c) of the truth values of $\alpha_1, \ldots, \alpha_n$

Example

$$\underbrace{1+1=5}_{\alpha_1}\ \underbrace{\text{or}}_{c}\ \underbrace{1+1\neq 5}_{\alpha_2}$$

is true if at least one of $1+1=5$, $1+1\neq 5$ is true

Propositional Logic (PL)

Simplest and most fundamental classical logic

All other classical logics are extensions of PL

Propositional Logic (PL)

Simplest and most fundamental classical logic

All other classical logics are extensions of PL

Propositional Logic Syntax: symbols

The set of symbols, or *alphabet*, of propositional logic consists of

1. a set $\mathcal{B}$ of *atomic symbols* or *atoms*:
   - truth constants: $\top$ (for true), $\bot$ (for false)
   - propositional variables: $p, q, r, \ldots$
2. *logical symbols:* connectives (i.e., $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$), parentheses (i.e., (,))

Propositional Logic Syntax: symbols

The set of symbols, or *alphabet*, of propositional logic consists of

1. a set $\mathcal{B}$ of *atomic symbols* or *atoms*:
   - truth constants: $\top$ (for true), $\bot$ (for false)
   - propositional variables: $p, q, r, \ldots$
2. *logical symbols:* connectives (i.e., $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$), parentheses (i.e., $($, $)$)

Propositional Logic Syntax: symbols

The set of symbols, or *alphabet*, of propositional logic consists of

1. a set $\mathcal{B}$ of *atomic symbols* or *atoms*:
   - truth constants: $\top$ (for true), $\bot$ (for false)
   - propositional variables: $p, q, r, \ldots$
2. *logical symbols:* connectives (i.e., $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$), parentheses (i.e., $($, $)$)

Note: We will use the same characters: '(' and ')' at three levels of discourse:

1. as part of propositional logic formulas, as in $(p \Rightarrow q)$
2. in mathematical notation, as in $f(x), \log(a)$
3. in regular text (as in here)

Propositional Logic Syntax: symbols

The set of symbols, or *alphabet*, of propositional logic consists of

1. a set $\mathcal{B}$ of *atomic symbols* or *atoms*:
   - truth constants: $\top$ (for true), $\bot$ (for false)
   - propositional variables: $p, q, r, \ldots$
2. *logical symbols:* connectives (i.e., $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$), parentheses (i.e., $($, $)$)

Note: We will use the same characters: '(' and ')' at three levels of discourse:

1. as part of propositional logic formulas, as in $(p \Rightarrow q)$
2. in mathematical notation, as in $f(x)$, $\log(a)$
3. in regular text (as in here)

Do not confuse the three!

Propositional Logic Syntax: expressions

A sentence, or formula, is a finite sequence of symbols

- $(p \wedge q)$
- $((\neg p) \Rightarrow r)$

Not all sequences of symbols are formulas:

- $(p \wedge \vee q)$
- pq
- $)) \Leftrightarrow)s$

Part of the syntax are rules that restrict formulas to a specific set of sequences

Propositional Logic Syntax: expressions

A sentence, or formula, is a finite sequence of symbols

- $(p \wedge q)$
- $((\neg p) \Rightarrow r)$

Not all sequences of symbols are formulas:

- $(p \wedge \vee q)$
- pq
- $)) \Leftrightarrow)s$

Part of the syntax are rules that restrict formulas to a specific set of sequences

Propositional Logic Syntax: expressions

A sentence, or formula, is a finite sequence of symbols

- $(p \wedge q)$
- $((\neg p) \Rightarrow r)$

Not all sequences of symbols are formulas:

- $(p \wedge \vee q)$
- pq
- $)) \Leftrightarrow)s$

Part of the syntax are rules that restrict formulas to a specific set of sequences

Propositional Logic Syntax: Formula-building operations

Consider the *formula-building operators* defined as follows for all formulas α and β:

- $\mathcal{E}_{\neg}(\alpha) = (\neg\alpha)$ (negation)
- $\mathcal{E}_{\wedge}(\alpha, \beta) = (\alpha \wedge \beta)$ (conjunction)
- $\mathcal{E}_{\vee}(\alpha, \beta) = (\alpha \vee \beta)$ (disjunction)
- $\mathcal{E}_{\Rightarrow}(\alpha, \beta) = (\alpha \Rightarrow \beta)$ (implication)
- $\mathcal{E}_{\Leftrightarrow}(\alpha, \beta) = (\alpha \Leftrightarrow \beta)$ (double implication)

Propositional Logic Syntax: Formula-building operations

Consider the *formula-building operators* defined as follows for all formulas α and β:

- $\mathcal{E}_{\neg}(\alpha) = (\neg\alpha)$ (negation)
- $\mathcal{E}_{\wedge}(\alpha, \beta) = (\alpha \wedge \beta)$ (conjunction)
- $\mathcal{E}_{\vee}(\alpha, \beta) = (\alpha \vee \beta)$ (disjunction)
- $\mathcal{E}_{\Rightarrow}(\alpha, \beta) = (\alpha \Rightarrow \beta)$ (implication)
- $\mathcal{E}_{\Leftrightarrow}(\alpha, \beta) = (\alpha \Leftrightarrow \beta)$ (double implication)

The set $\mathcal{W}$ of *well-formed formulas*, or simply *formulas* or *wffs*, is the set of all sentences finitely-generated by the operators above from the atoms in $\mathcal{B}$

Propositional Logic Syntax: Formula-building operations

Consider the *formula-building operators* defined as follows for all formulas α and β:

- $\mathcal{E}_{\neg}(\alpha) = (\neg\alpha)$ (negation)
- $\mathcal{E}_{\wedge}(\alpha, \beta) = (\alpha \wedge \beta)$ (conjunction)
- $\mathcal{E}_{\vee}(\alpha, \beta) = (\alpha \vee \beta)$ (disjunction)
- $\mathcal{E}_{\Rightarrow}(\alpha, \beta) = (\alpha \Rightarrow \beta)$ (implication)
- $\mathcal{E}_{\Leftrightarrow}(\alpha, \beta) = (\alpha \Leftrightarrow \beta)$ (double implication)

In other words,

- every atom in $\mathcal{B}$ is a wff
- if α and β are wffs, so are the expressions generated from them by $\mathcal{E}_{\neg}$, $\mathcal{E}_{\wedge}$, $\mathcal{E}_{\vee}$, $\mathcal{E}_{\Rightarrow}$, and $\mathcal{E}_{\Leftrightarrow}$
- nothing else is a wff

Closed sets and generated sets

A set S is *closed under* a set F of operators if applying any of those operators to elements of S results in an element that is also in S

Closed sets and generated sets

A set S is *closed under* a set F of operators if applying any of those operators to elements of S results in an element that is also in S

Examples

- The set $\mathbb{N}$ of all natural numbers is closed under addition and multiplication but not negation
- The set $\mathbb{Z}$ of all integer numbers is closed under addition, multiplication, and negation
- The set $\mathbb{E}$ of all even integers is closed under addition, multiplication, and negation
- The set $\mathbb{O}$ of all odd integers is closed under multiplication and negation but not under addition

Closed sets and generated sets

A set S is *closed under* a set F of operators if applying any of those operators to elements of S results in an element that is also in S

A set C *is generated* from a set B by a set F of operators if it is the smallest set that is closed under F and contains B

Closed sets and generated sets

A set S is *closed under* a set F of operators if applying any of those operators to elements of S results in an element that is also in S

A set C *is generated* from a set B by a set F of operators if it is the smallest set that is closed under F and contains B

Examples

- The set $\mathbb{N}$ of all natural numbers is generated from $\{0, 1\}$ by $\{+\}$
- The set $\mathbb{Z}$ of all integer numbers is generated from $\{1\}$ by $\{+, -\}$
- The set $\mathbb{E}$ of all even integers is generated from $\{2\}$ by $\{+, -\}$
- The set $\mathbb{R}$ of all real number is generated from no sets of numbers[1]

[1] Generated sets are necessarily countable.

The Structural Induction Principle

Consider a set C generated from a set B by a set F of operators

If a set S includes B and is closed under F, we say S is *inductive with respect to C*

The Structural Induction Principle

Consider a set C generated from a set B by a set F of operators

If a set S includes B and is closed under F, we say S is *inductive with respect to* C

Example $\mathbb{Z}$ is inductive w.r.t. $\mathbb{N}$ (which is generated from $\{0, 1\}$ by $\{+\}$)

The Structural Induction Principle

Consider a set C generated from a set B by a set F of operators

If a set S includes B and is closed under F, we say S is *inductive with respect to* C

Note: S inductive w.r.t. C implies that $C \subseteq S$

The Structural Induction Principle

Consider a set C generated from a set B by a set F of operators

If a set S includes B and is closed under F, we say S is *inductive with respect to* C

We can use the structural induction principle to show that a set like C above has a particular property P

The Structural Induction Principle

Consider a set C generated from a set B by a set F of operators

If a set S includes B and is closed under F, we say S is *inductive with respect to* C

We can use the structural induction principle to show that a set like C above has a particular property P

The argument goes like this:

1. Consider a set S whose elements all have property P
2. Show that S is inductive with respect to C

This proves that $C \subseteq S$ and thus all elements of C have property P

The Structural Induction Principle

Consider a set C generated from a set B by a set F of operators

If a set S includes B and is closed under F, we say S is *inductive with respect to* C

We can use the structural induction principle to show that a set like C above has a particular property P

We often use structural induction to prove properties about formulas

Structural Induction: Example

Given our inductive definition of well-formed formulas,
we can use the induction principle to prove things about the set $\mathcal{W}$ of wffs

Example

Prove that every wff has the same number of left parentheses and right parentheses

Proof

Let $l(\alpha)$ be the number of left parentheses and
let $r(\alpha)$ be the number of right parentheses in an expression α

Let S be the set of all expressions α such that $l(\alpha) = r(\alpha)$

We wish to show that $\mathcal{W} \subseteq S$

This follows from the induction principle if we can show that S is inductive w.r.t. $\mathcal{W}$

Structural Induction: Example

Given our inductive definition of well-formed formulas,
we can use the induction principle to prove things about the set $\mathcal{W}$ of wffs

Example

Prove that every wff has the same number of left parentheses and right parentheses

Proof

Let $l(\alpha)$ be the number of left parentheses and
let $r(\alpha)$ be the number of right parentheses in an expression α

Let S be the set of all expressions α such that $l(\alpha) = r(\alpha)$

We wish to show that $\mathcal{W} \subseteq S$

This follows from the induction principle if we can show that S is inductive w.r.t. $\mathcal{W}$

Structural Induction: Example

Given our inductive definition of well-formed formulas,
we can use the induction principle to prove things about the set $\mathcal{W}$ of wffs

Example

Prove that every wff has the same number of left parentheses and right parentheses

Proof

Let $l(\alpha)$ be the number of left parentheses and
let $r(\alpha)$ be the number of right parentheses in an expression α

Let S be the set of all expressions α such that $l(\alpha) = r(\alpha)$

We wish to show that $\mathcal{W} \subseteq S$

This follows from the induction principle if we can show that S is inductive w.r.t. $\mathcal{W}$

Structural Induction: Example (cont.)

Base Case:

We must show that $\mathcal{B} \subseteq S$

Recall that $\mathcal{B}$ is the set of expressions consisting of a single propositional symbol

It is clear that for such expressions, $l(\alpha) = r(\alpha) = 0$

Structural Induction: Example (cont.)

Inductive Case:

We must show that S is closed under each formula-building operator

- $\mathcal{E}_\neg$
  Let $\alpha \in S$. We know that $\mathcal{E}_\neg(\alpha) = (\neg\alpha)$.
  It follows that $l(\mathcal{E}_\neg(\alpha)) = 1 + l(\alpha)$ and $r(\mathcal{E}_\neg(\alpha)) = 1 + r(\alpha)$.
  Since $\alpha \in S$, we know that $l(\alpha) = r(\alpha)$; it follows that $l(\mathcal{E}_\neg(\alpha)) = r(\mathcal{E}_\neg(\alpha))$, and thus $\mathcal{E}_\neg(\alpha) \in S$.

- $\mathcal{E}_\wedge$
  Let $\alpha, \beta \in S$. We know that $\mathcal{E}_\wedge(\alpha, \beta) = (\alpha \wedge \beta)$.
  Thus $l(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + l(\alpha) + l(\beta)$ and $r(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + r(\alpha) + r(\beta)$.
  As before, it follows from the inductive hypothesis that $\mathcal{E}_\wedge(\alpha, \beta) \in S$

- The arguments for $\mathcal{E}_\vee$, $\mathcal{E}_\rightarrow$, and $\mathcal{E}_\leftrightarrow$ are analogous to the one for $\mathcal{E}_\wedge$.

Structural Induction: Example (cont.)

Inductive Case:

We must show that S is closed under each formula-building operator

- $\mathcal{E}_\neg$
  Let $\alpha \in S$. We know that $\mathcal{E}_\neg(\alpha) = (\neg\alpha)$.
  It follows that $l(\mathcal{E}_\neg(\alpha)) = 1 + l(\alpha)$ and $r(\mathcal{E}_\neg(\alpha)) = 1 + r(\alpha)$.
  Since $\alpha \in S$, we know that $l(\alpha) = r(\alpha)$; it follows that $l(\mathcal{E}_\neg(\alpha)) = r(\mathcal{E}_\neg(\alpha))$, and thus $\mathcal{E}_\neg(\alpha) \in S$.

- $\mathcal{E}_\wedge$
  Let $\alpha, \beta \in S$. We know that $\mathcal{E}_\wedge(\alpha, \beta) = (\alpha \wedge \beta)$.
  Thus $l(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + l(\alpha) + l(\beta)$ and $r(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + r(\alpha) + r(\beta)$.
  As before, it follows from the inductive hypothesis that $\mathcal{E}_\wedge(\alpha, \beta) \in S$

- The arguments for $\mathcal{E}_\vee$, $\mathcal{E}_\rightarrow$, and $\mathcal{E}_\leftrightarrow$ are analogous to the one for $\mathcal{E}_\wedge$.

Structural Induction: Example (cont.)

Inductive Case:

We must show that S is closed under each formula-building operator

- $\mathcal{E}_\neg$
  Let $\alpha \in S$. We know that $\mathcal{E}_\neg(\alpha) = (\neg\alpha)$.
  It follows that $l(\mathcal{E}_\neg(\alpha)) = 1 + l(\alpha)$ and $r(\mathcal{E}_\neg(\alpha)) = 1 + r(\alpha)$.
  Since $\alpha \in S$, we know that $l(\alpha) = r(\alpha)$; it follows that $l(\mathcal{E}_\neg(\alpha)) = r(\mathcal{E}_\neg(\alpha))$, and thus $\mathcal{E}_\neg(\alpha) \in S$.

- $\mathcal{E}_\wedge$
  Let $\alpha, \beta \in S$. We know that $\mathcal{E}_\wedge(\alpha, \beta) = (\alpha \wedge \beta)$.
  Thus $l(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + l(\alpha) + l(\beta)$ and $r(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + r(\alpha) + r(\beta)$.
  As before, it follows from the inductive hypothesis that $\mathcal{E}_\wedge(\alpha, \beta) \in S$

- The arguments for $\mathcal{E}_\vee$, $\mathcal{E}_\rightarrow$, and $\mathcal{E}_\leftrightarrow$ are analogous to the one for $\mathcal{E}_\wedge$.

Structural Induction: Example (cont.)

Inductive Case:

We must show that S is closed under each formula-building operator

- $\mathcal{E}_\neg$
  Let $\alpha \in S$. We know that $\mathcal{E}_\neg(\alpha) = (\neg\alpha)$.
  It follows that $l(\mathcal{E}_\neg(\alpha)) = 1 + l(\alpha)$ and $r(\mathcal{E}_\neg(\alpha)) = 1 + r(\alpha)$.
  Since $\alpha \in S$, we know that $l(\alpha) = r(\alpha)$; it follows that $l(\mathcal{E}_\neg(\alpha)) = r(\mathcal{E}_\neg(\alpha))$, and thus $\mathcal{E}_\neg(\alpha) \in S$.

- $\mathcal{E}_\wedge$
  Let $\alpha, \beta \in S$. We know that $\mathcal{E}_\wedge(\alpha, \beta) = (\alpha \wedge \beta)$.
  Thus $l(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + l(\alpha) + l(\beta)$ and $r(\mathcal{E}_\wedge(\alpha, \beta)) = 1 + r(\alpha) + r(\beta)$.
  As before, it follows from the inductive hypothesis that $\mathcal{E}_\wedge(\alpha, \beta) \in S$

- The arguments for $\mathcal{E}_\vee$, $\mathcal{E}_\rightarrow$, and $\mathcal{E}_\leftrightarrow$ are analogous to the one for $\mathcal{E}_\wedge$.

Notational conventions for formulas

- We fix a countably infinite set of propositional variables
  We typically use $p, q, r, p_1, p_2, p_3, \ldots$ to denote them

- We may omit outermost parentheses, e.g., write $p \wedge q$ instead of $(p \wedge q)$

- We may further omit parentheses by defining *order of operations (precedence)*:
  - Negation binds most strongly, with small as possible scope: $\neg p \wedge q$ means $((\neg p) \wedge q)$
  - $\wedge$ binds more strongly than $\vee$: $p_1 \wedge p_2 \vee p_3$ means $(p_1 \wedge p_2) \vee p_3$
  - $\vee$ binds more strongly than $\Rightarrow$, $\Leftrightarrow$: $p_1 \wedge p_2 \Rightarrow \neg p_3 \vee p_4$ means $(p_1 \wedge p_2) \Rightarrow (\neg p_3 \vee p_4)$
  - Binary connectives are treated as right-associative: $p_1 \wedge p_2 \wedge p_3$ means $p_1 \wedge (p_2 \wedge p_3)$

- We use $\alpha, \beta, \gamma, \varphi, \psi$ to denote arbitrary wffs

Notational conventions for formulas

- We fix a countably infinite set of propositional variables
  We typically use $p, q, r, p_1, p_2, p_3, \ldots$ to denote them
- We may omit outermost parentheses, e.g., write $p \wedge q$ instead of $(p \wedge q)$
- We may further omit parentheses by defining *order of operations (precedence)*:
  - Negation binds most strongly, with small as possible scope: $\neg p \wedge q$ means $((\neg p) \wedge q)$
  - $\wedge$ binds more strongly than $\vee$: $p_1 \wedge p_2 \vee p_3$ means $(p_1 \wedge p_2) \vee p_3$
  - $\vee$ binds more strongly than $\Rightarrow$, $\Leftrightarrow$: $p_1 \wedge p_2 \Rightarrow \neg p_3 \vee p_4$ means $(p_1 \wedge p_2) \Rightarrow (\neg p_3 \vee p_4)$
  - Binary connectives are treated as right-associative: $p_1 \wedge p_2 \wedge p_3$ means $p_1 \wedge (p_2 \wedge p_3)$
- We use $\alpha, \beta, \gamma, \varphi, \psi$ to denote arbitrary wffs

Notational conventions for formulas

- We fix a countably infinite set of propositional variables
  We typically use $p, q, r, p_1, p_2, p_3, \ldots$ to denote them

- We may omit outermost parentheses, e.g., write $p \land q$ instead of $(p \land q)$

- We may further omit parentheses by defining *order of operations (precedence)*:
  - Negation binds most strongly, with small as possible scope: $\neg p \land q$ means $((\neg p) \land q)$
  - $\land$ binds more strongly than $\lor$: $p_1 \land p_2 \lor p_3$ means $(p_1 \land p_2) \lor p_3$
  - $\lor$ binds more strongly than $\Rightarrow$, $\Leftrightarrow$: $p_1 \land p_2 \Rightarrow \neg p_3 \lor p_4$ means $(p_1 \land p_2) \Rightarrow (\neg p_3 \lor p_4)$
  - Binary connectives are treated as right-associative: $p_1 \land p_2 \land p_3$ means $p_1 \land (p_2 \land p_3)$

- We use $\alpha, \beta, \gamma, \varphi, \psi$ to denote arbitrary wffs

Notational conventions for formulas

- We fix a countably infinite set of propositional variables
  We typically use $p, q, r, p_1, p_2, p_3, \ldots$ to denote them
- We may omit outermost parentheses, e.g., write $p \wedge q$ instead of $(p \wedge q)$
- We may further omit parentheses by defining *order of operations (precedence)*:
  - Negation binds most strongly, with small as possible scope: $\neg p \wedge q$ means $((\neg p) \wedge q)$
  - $\wedge$ binds more strongly than $\vee$: $p_1 \wedge p_2 \vee p_3$ means $(p_1 \wedge p_2) \vee p_3$
  - $\vee$ binds more strongly than $\Rightarrow$, $\Leftrightarrow$: $p_1 \wedge p_2 \Rightarrow \neg p_3 \vee p_4$ means $(p_1 \wedge p_2) \Rightarrow (\neg p_3 \vee p_4)$
  - Binary connectives are treated as right-associative: $p_1 \wedge p_2 \wedge p_3$ means $p_1 \wedge (p_2 \wedge p_3)$
- We use $\alpha, \beta, \gamma, \varphi, \psi$ to denote arbitrary wffs

Notational conventions for formulas

- We fix a countably infinite set of propositional variables
  We typically use $p, q, r, p_1, p_2, p_3, \ldots$ to denote them

- We may omit outermost parentheses, e.g., write $p \wedge q$ instead of $(p \wedge q)$

- We may further omit parentheses by defining *order of operations (precedence)*:
  - Negation binds most strongly, with small as possible scope: $\neg p \wedge q$ means $((\neg p) \wedge q)$
  - $\wedge$ binds more strongly than $\vee$: $p_1 \wedge p_2 \vee p_3$ means $(p_1 \wedge p_2) \vee p_3$
  - $\vee$ binds more strongly than $\Rightarrow$, $\Leftrightarrow$: $p_1 \wedge p_2 \Rightarrow \neg p_3 \vee p_4$ means $(p_1 \wedge p_2) \Rightarrow (\neg p_3 \vee p_4)$
  - Binary connectives are treated as right-associative: $p_1 \wedge p_2 \wedge p_3$ means $p_1 \wedge (p_2 \wedge p_3)$

- We use $\alpha, \beta, \gamma, \varphi, \psi$ to denote arbitrary wffs

Propositional Logic: Compositional Semantics

The meaning of a wff α is a truth value: true or false

Given a mapping v from the propositional variables in α to $\{$ false, true $\}$, the meaning of α is depends on the meaning of its subformulas

The mapping v is a *variable assignment*, or *interpretation*, of (the variables of) α

Propositional Logic: Compositional Semantics

The meaning of a wff α is a truth value: true or false

Given a mapping *v* from the propositional variables in α to $\{$ false, true $\}$, the meaning of α is depends on the meaning of its subformulas

The mapping *v* is a *variable assignment*, or *interpretation*, of (the variables of) α

Propositional Logic: Compositional Semantics

The meaning of a wff α is a truth value: true or false

Given a mapping v from the propositional variables in α to $\{ \text{false}, \text{true} \}$, the meaning of α is depends on the meaning of its subformulas

The mapping v is a *variable assignment*, or *interpretation*, of (the variables of) α

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \rightarrow \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \land \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \lor \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

We can extend v to an interpretation $\overline{v} : \mathcal{W} \to \{\text{false}, \text{true}\}$ for the whole set of $\mathcal{W}$, defined by structural induction on wffs as follows:

- $\overline{v}(\bot) = \text{false}$ and $\overline{v}(\top) = \text{true}$
- $\overline{v}(p) = v(p)$ for all propositional variables p
- $\overline{v}(\neg\alpha) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$
- $\overline{v}(\alpha \wedge \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \vee \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{true}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Rightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \text{false}$ or $\overline{v}(\beta) = \text{true}$
- $\overline{v}(\alpha \Leftrightarrow \beta) = \text{true}$ iff $\overline{v}(\alpha) = \overline{v}(\beta)$

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

For every $\alpha \in \mathcal{W}$, we will use the following statements interchangeably

- $v \models \alpha$
- $\overline{v}(\alpha) = \text{true}$
- v is a *model* of α
- v is a *satisfying assignment* of α
- v *satisfies* α

Propositional Logic: Semantics

Let v be a variable assignment for all the propositional variables of $\mathcal{B}$

For every $\alpha \in \mathcal{W}$, we will use the following statements interchangeably

- $v \models \alpha$
- $\overline{v}(\alpha) = \text{true}$
- v is a *model* of α
- v is a *satisfying assignment* of α
- v *satisfies* α

Satisfiability of formulas

A wff α is *satisfiable*
  if $\overline{v}(\alpha) = \text{true}$ for some interpretation v

A wff α is *falsifiable*
  if $\overline{v}(\alpha) = \text{false}$ for some interpretation v

A wff α is *unsatisfiable*
  if it is not satisfiable, i.e., $\overline{v}(\alpha) = \text{false}$ for all interpretations v

A set $U \subseteq \mathcal{W}$ is *(un)satisfiable*
  if there is (no) interpretation v such that $\overline{v}(\alpha) = \text{true}$ for all $\alpha \in U$

Satisfiability of formulas

A wff α is *satisfiable*
  if $\overline{v}(\alpha) = \text{true}$ for some interpretation v

A wff α is *falsifiable*
  if $\overline{v}(\alpha) = \text{false}$ for some interpretation v

A wff α is *unsatisfiable*
  if it is not satisfiable, i.e., $\overline{v}(\alpha) = \text{false}$ for all interpretations v

A set $U \subseteq \mathcal{W}$ is *(un)satisfiable*
  if there is (no) interpretation v such that $\overline{v}(\alpha) = \text{true}$ for all $\alpha \in U$

Satisfiability of formulas

A wff α is *satisfiable*
if $\overline{v}(\alpha) = \text{true}$ for some interpretation v

A wff α is *falsifiable*
if $\overline{v}(\alpha) = \text{false}$ for some interpretation v

A wff α is *unsatisfiable*
if it is not satisfiable, i.e., $\overline{v}(\alpha) = \text{false}$ for all interpretations v

A set $U \subseteq \mathcal{W}$ is *(un)satisfiable*
if there is (no) interpretation v such that $\overline{v}(\alpha) = \text{true}$ for all $\alpha \in U$

Satisfiability of formulas

A wff α is *satisfiable*
if $\overline{v}(\alpha) = \text{true}$ for some interpretation v

A wff α is *falsifiable*
if $\overline{v}(\alpha) = \text{false}$ for some interpretation v

A wff α is *unsatisfiable*
if it is not satisfiable, i.e., $\overline{v}(\alpha) = \text{false}$ for all interpretations v

A set $U \subseteq \mathcal{W}$ is *(un)satisfiable*
if there is (no) interpretation v such that $\overline{v}(\alpha) = \text{true}$ for all $\alpha \in U$

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \vDash \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \vDash \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \models \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Special cases:

- If $\varnothing \models \alpha$, α is a *tautology* or is *valid* and write $\models \alpha$
- α_1, α_2 are *logically equivalent*, written $\alpha_1 \equiv \alpha_2$, iff $\{\alpha_1\} \models \alpha_2$ and $\{\alpha_2\} \models \alpha_1$
- We write $\alpha \models \beta$ as a shorthand for $\{\alpha\} \models \beta$

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \models \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Special cases:

- If $\varnothing \models \alpha$, α is a *tautology* or is *valid* and write $\models \alpha$
- α_1, α_2 are *logically equivalent*, written $\alpha_1 \equiv \alpha_2$, iff $\{\alpha_1\} \models \alpha_2$ and $\{\alpha_2\} \models \alpha_1$
- We write $\alpha \models \beta$ as a shorthand for $\{\alpha\} \models \beta$

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \models \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Special cases:

- If $\varnothing \models \alpha$, α is a *tautology* or is *valid* and write $\models \alpha$
- α_1, α_2 are *logically equivalent*, written $\alpha_1 \equiv \alpha_2$, iff $\{\alpha_1\} \models \alpha_2$ and $\{\alpha_2\} \models \alpha_1$
- We write $\alpha \models \beta$ as a shorthand for $\{\alpha\} \models \beta$

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \models \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Special cases:

- If $\varnothing \models \alpha$, α is a *tautology* or is *valid* and write $\models \alpha$
- α_1, α_2 are *logically equivalent*, written $\alpha_1 \equiv \alpha_2$, iff $\{ \alpha_1 \} \models \alpha_2$ and $\{ \alpha_2 \} \models \alpha_1$
- We write $\alpha \models \beta$ as a shorthand for $\{\alpha\} \models \beta$

Logical implication and validity

A set $U \subseteq \mathcal{W}$ *entails* or *logically implies* a wff β, written $U \vDash \beta$, if every satisfying assignment v for U satisfies β as well

We also say that U *entails* β and β is a *logical consequence* of U

Note: We use $\vDash$ for two different relations:

1. satisfaction between a variable assignment and a formula ($\overline{v} \vDash \alpha$)
2. entailment between a set of formulas and a formula ($\{\alpha_1, \alpha_2, \ldots\} \vDash \alpha$)

Use context to disambiguate!

Satisfiability vs. validity

Satisfiability and validity are dual concepts:

a wff α is valid iff $\neg\alpha$ is unsatisfiable

Consequence:

If we have a procedure that can check satisfiability,
then we can also check validity, and vice versa

Satisfiability vs. validity

Satisfiability and validity are dual concepts:

a wff α is valid iff $\neg\alpha$ is unsatisfiable

Consequence:

If we have a procedure that can check satisfiability,
then we can also check validity, and vice versa

Examples

p, q propositional variables α, β, γ formulas

- $p, \ p \Rightarrow q, \ p \vee \neg q, \ (p \Rightarrow q) \Rightarrow p$ are all satisfiable
- $p, \ p \Rightarrow q, \ p \vee \neg q, \ (p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha, \ \alpha \vee \neg \alpha, \ \alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \vDash \alpha, \ \alpha \wedge \beta \vDash \beta, \ \{\alpha, \alpha \Rightarrow \beta\} \vDash \beta, \ \{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \vDash \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Examples

p, q propositional variables α, β, γ formulas

- p, $p \Rightarrow q$, $p \vee \neg q$, $(p \Rightarrow q) \Rightarrow p$ are all satisfiable
- p, $p \Rightarrow q$, $p \vee \neg q$, $(p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha$, $\alpha \vee \neg\alpha$, $\alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \models \alpha$, $\alpha \wedge \beta \models \beta$, $\{\alpha, \alpha \Rightarrow \beta\} \models \beta$, $\{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \models \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Examples

p, q propositional variables $\alpha,\ \beta,\ \gamma$ formulas

- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all satisfiable
- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha,\ \alpha \vee \neg\alpha,\ \alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \vDash \alpha,\ \alpha \wedge \beta \vDash \beta,\ \{\alpha, \alpha \Rightarrow \beta\} \vDash \beta,\ \{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \vDash \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Examples

p, q propositional variables α, β, γ formulas

- p, $p \Rightarrow q$, $p \vee \neg q$, $(p \Rightarrow q) \Rightarrow p$ are all satisfiable
- p, $p \Rightarrow q$, $p \vee \neg q$, $(p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha$, $\alpha \vee \neg\alpha$, $\alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \models \alpha$, $\alpha \wedge \beta \models \beta$, $\{\alpha, \alpha \Rightarrow \beta\} \models \beta$, $\{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \models \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Examples

p, q propositional variables $\qquad \alpha, \beta, \gamma$ formulas

- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all satisfiable
- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha,\ \alpha \vee \neg\alpha,\ \alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \vDash \alpha,\ \alpha \wedge \beta \vDash \beta,\ \{\alpha, \alpha \Rightarrow \beta\} \vDash \beta,\ \{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \vDash \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Examples

p, q propositional variables α, β, γ formulas

- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all satisfiable
- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha,\ \alpha \vee \neg \alpha,\ \alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \models \alpha,\ \alpha \wedge \beta \models \beta,\ \{\alpha, \alpha \Rightarrow \beta\} \models \beta,\ \{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \models \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Examples

p, q propositional variables α, β, γ formulas

- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all satisfiable
- $p,\ p \Rightarrow q,\ p \vee \neg q,\ (p \Rightarrow q) \Rightarrow p$ are all falsifiable
- $\alpha \Rightarrow \alpha,\ \alpha \vee \neg\alpha,\ \alpha \Rightarrow (\beta \Rightarrow \alpha)$ are all valid
- $\alpha \vDash \alpha,\ \alpha \wedge \beta \vDash \beta,\ \{\alpha, \alpha \Rightarrow \beta\} \vDash \beta,\ \{\alpha, \beta, (\alpha \vee \beta) \Rightarrow \gamma\} \vDash \gamma$

Note:

- $\top$ is valid and $\bot$ is unsatisfiable
- Every valid formula is satisfiable but not falsifiable
- Every unsatisfiable formula is falsifiable

Implication ($\Rightarrow$) vs. logical implication ($\models$)

The two concepts are semantically related:

$$\alpha \models \beta \quad \text{iff} \quad \models \alpha \Rightarrow \beta$$

Implication ($\Rightarrow$) vs. logical implication ($\models$)

The two concepts are semantically related:

$$\alpha \models \beta \quad \text{iff} \quad \models \alpha \Rightarrow \beta$$

Proof: Exercise

Implication ($\Rightarrow$) vs. logical implication ($\models$)

The two concepts are semantically related:

$$\alpha \models \beta \quad \text{iff} \quad \models \alpha \Rightarrow \beta$$

Correspondingly:

$$\alpha \equiv \beta \quad \text{iff} \quad \models \alpha \Leftrightarrow \beta$$

Implication ($\Rightarrow$) vs. logical implication ($\models$)

The two concepts are semantically related:

$$\alpha \models \beta \quad \text{iff} \quad \models \alpha \Rightarrow \beta$$

Correspondingly:

$$\alpha \equiv \beta \quad \text{iff} \quad \models \alpha \Leftrightarrow \beta$$

because

$$\alpha \equiv \beta \quad \text{iff} \quad \alpha \models \beta \text{ and } \beta \models \alpha$$

and

$$\models \alpha \Leftrightarrow \beta \quad \text{iff} \quad \models \alpha \Rightarrow \beta \text{ and} \models \beta \Rightarrow \alpha$$

Implication ($\Rightarrow$) vs. logical implication ($\models$)

The two concepts are semantically related:

$$\alpha \models \beta \quad \text{iff} \quad \models \alpha \Rightarrow \beta$$

Correspondingly:

$$\alpha \equiv \beta \quad \text{iff} \quad \models \alpha \Leftrightarrow \beta$$

Note: $\alpha \models \beta$ and $\alpha \equiv \beta$ are mathematical statements, *not formulas*

Defining One Operator in Terms of Another

A binary connective $\circ$ over wffs is *defined from* a set of connectives C
if for all wffs α and β, $\alpha \circ \beta \equiv \gamma$,
where γ is constructed by applying only connectives in C to α and β

Defining One Operator in Terms of Another

A binary connective $\circ$ over wffs is *defined from* a set of connectives C
if for all wffs α and β, $\alpha \circ \beta \equiv \gamma$,
where γ is constructed by applying only connectives in C to α and β

The connectives $\vee, \wedge, \Rightarrow, \Leftrightarrow$ can be defined from $\neg$ and one of $\vee, \wedge, \Rightarrow, \Leftrightarrow$

Defining One Operator in Terms of Another

A binary connective $\circ$ over wffs is *defined from* a set of connectives C
if for all wffs α and β, $\alpha \circ \beta \equiv \gamma$,
where γ is constructed by applying only connectives in C to α and β

The connectives $\vee, \wedge, \Rightarrow, \Leftrightarrow$ can be defined from $\neg$ and one of $\vee, \wedge, \Rightarrow, \Leftrightarrow$

Example: defining $\vee, \wedge, \Leftrightarrow$ from $\{\neg, \Rightarrow\}$

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

Defining One Operator in Terms of Another

A binary connective $\circ$ over wffs is *defined from* a set of connectives C
if for all wffs α and β, $\alpha \circ \beta \equiv \gamma$,
where γ is constructed by applying only connectives in C to α and β

The connectives $\vee, \wedge, \Rightarrow, \Leftrightarrow$ can be defined from $\neg$ and one of $\vee, \wedge, \Rightarrow, \Leftrightarrow$

Why do we care about this?

- To simplify arguments by structural induction
- Many algorithms are defined over normal forms using a specified subset of connectives

Decision Procedure in Propositional Logic

Let $U \in \mathcal{W}$

A *decision procedure* for U is a terminating procedure[2] that takes wffs as input and for each input α returns

yes if $\alpha \in U$ no if $\alpha \notin U$

This course: We consider decision procedures for validity/satisfiability, that is, U will the set of valid/satisfiable formulas

[2] A procedure does not necessarily terminate, whereas an algorithm does, by definition

Decision Procedure in Propositional Logic

Let $U \in \mathcal{W}$

A *decision procedure* for U is a terminating procedure[2] that takes wffs as input and for each input α returns

yes if $\alpha \in U$ no if $\alpha \notin U$

This course: We consider decision procedures for validity/satisfiability, that is, U will the set of valid/satisfiable formulas

[2] A procedure does not necessarily terminate, whereas an algorithm does, by definition

Basic Decision Procedures for Validity/Satisfiability

Two fundamental strategies for deciding validity/satisfiability:

- *Search-based procedures*:
  search the space of possible interpretations of the given wff
- *Deduction-based procedures*:
  use an inference system based on axioms and inference rules to deduce validity

SAT solvers (covered later) interleave search and deduction

Basic Decision Procedures for Validity/Satisfiability

Two fundamental strategies for deciding validity/satisfiability:

- *Search-based procedures*:
  search the space of possible interpretations of the given wff
- *Deduction-based procedures*:
  use an inference system based on axioms and inference rules to deduce validity

SAT solvers (covered later) interleave search and deduction

Basic Decision Procedures for Validity/Satisfiability

Two fundamental strategies for deciding validity/satisfiability:

- *Search-based procedures*:
  search the space of possible interpretations of the given wff
- *Deduction-based procedures*:
  use an inference system based on axioms and inference rules to deduce validity

SAT solvers (covered later) interleave search and deduction

Basic Decision Procedures for Validity/Satisfiability

Two fundamental strategies for deciding validity/satisfiability:

- *Search-based procedures*:
  search the space of possible interpretations of the given wff
- *Deduction-based procedures*:
  use an inference system based on axioms and inference rules to deduce validity

SAT solvers (covered later) interleave search and deduction

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \land q) \Rightarrow (p \lor \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \land q$	$\neg q$	$p \lor \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \land q) \Rightarrow (p \lor \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \land q$	$\neg q$	$p \lor \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

The Truth-table Method

In PL, it is possible to enumerate all the interpretations, e.g., with *truth tables*

Example: is $\alpha := (p \wedge q) \Rightarrow (p \vee \neg q)$ a valid formula?

Writing 0 for false and 1 for true, for conciseness:

p	q	$p \wedge q$	$\neg q$	$p \vee \neg q$	α
0	0	0	1	1	1
0	1	0	0	0	1
1	0	0	1	1	1
1	1	1	0	1	1

Drawbacks?

- Need to evaluate a formula for each of 2^n possible interpretations
  This can be memory efficient but is runtime inefficient
- Works because the number of interpretations of a formula is finite

Proof by deduction

Informally, a *proof system* consists of a set of *proof rules*

A proof rule consists of:

- *premises* (or antecedents): facts that must hold for the rule apply
- *conclusions* (or consequents): facts deduced/derived from applying the rule

$$\frac{P_1 \quad \cdots \quad P_n}{C_{1,1}, \ldots, C_{1,n_1} \mid \cdots \mid C_{m,1}, \ldots, C_{m,n_m}}$$

Proof by deduction

Informally, a *proof system* consists of a set of *proof rules*

A proof rule consists of:

- *premises* (or antecedents): facts that must hold for the rule apply
- *conclusions* (or consequents): facts deduced/derived from applying the rule

$$\frac{P_1 \quad \cdots \quad P_n}{C_{1,1}, \ldots, C_{1,n_1} \;\big|\; \cdots \;\big|\; C_{m,1}, \ldots, C_{m,n_m}}$$

$$\frac{\alpha \qquad \beta}{\alpha \wedge \beta} \qquad \frac{\alpha \qquad \alpha \Rightarrow \beta}{\beta} \qquad \frac{\alpha \Leftrightarrow \beta}{\alpha, \beta \;\big|\; \neg\alpha, \neg\beta}$$

Proof by deduction

Informally, a *proof system* consists of a set of *proof rules*

A proof rule consists of:

- *premises* (or antecedents): facts that must hold for the rule apply
- *conclusions* (or consequents): facts deduced/derived from applying the rule

$$\frac{P_1 \quad \cdots \quad P_n}{C_{1,1}, \ldots, C_{1,n_1} \;\Big|\; \cdots \;\Big|\; C_{m,1}, \ldots, C_{m,n_m}}$$

Commas indicate derivation of multiple conclusions

Pipes indicate alternative conclusions (giving rise to *proof branches*)

$$\frac{\alpha \quad \beta}{\alpha \wedge \beta} \qquad \frac{\alpha \quad \alpha \Rightarrow \beta}{\beta} \qquad \frac{\alpha \Leftrightarrow \beta}{\alpha, \beta \;\Big|\; \neg\alpha, \neg\beta}$$

Proof by deduction

Informally, a *proof system* consists of a set of *proof rules*

A proof rule consists of:

- *premises* (or antecedents): facts that must hold for the rule apply
- *conclusions* (or consequents): facts deduced/derived from applying the rule

$$\frac{P_1 \quad \cdots \quad P_n}{C_{1,1}, \ldots, C_{1,n_1} \;\middle|\; \cdots \;\middle|\; C_{m,1}, \ldots, C_{m,n_m}}$$

Examples:

$$\frac{\alpha \qquad \beta}{\alpha \wedge \beta} \qquad\qquad \frac{\alpha \qquad \alpha \Rightarrow \beta}{\beta} \qquad\qquad \frac{\alpha \Leftrightarrow \beta}{\alpha,\ \beta \mid \neg\alpha, \neg\beta}$$

Proof by deduction: semantic arguments

Premises and conclusions can be anything

including satisfiability assertions about some interpretation v

$$\frac{v \models \neg\alpha}{v \not\models \alpha}$$

$$\frac{v \not\models \neg\alpha}{v \models \alpha}$$

$$\frac{v \models \alpha \land \beta}{v \models \alpha,\ v \models \beta}$$

$$\frac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$$

$$\frac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$$

$$\frac{v \not\models \alpha \lor \beta}{v \not\models \alpha,\ v \not\models \beta}$$

$$\frac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$$

$$\frac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha,\ v \not\models \beta}$$

$$\frac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha,\ v \models \beta \mid v \not\models \alpha,\ v \not\models \beta}$$

$$\frac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha,\ v \models \beta \mid v \models \alpha,\ v \not\models \beta}$$

$$\frac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$$

Proof by deduction: semantic arguments

Premises and conclusions can be anything

including satisfiability assertions about some interpretation v

$$\frac{v \models \neg\alpha}{v \not\models \alpha}$$

$$\frac{v \not\models \neg\alpha}{v \models \alpha}$$

$$\frac{v \models \alpha \land \beta}{v \models \alpha,\ v \models \beta}$$

$$\frac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$$

$$\frac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$$

$$\frac{v \not\models \alpha \lor \beta}{v \not\models \alpha,\ v \not\models \beta}$$

$$\frac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$$

$$\frac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha,\ v \not\models \beta}$$

$$\frac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha,\ v \models \beta \mid v \not\models \alpha,\ v \not\models \beta}$$

$$\frac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha,\ v \models \beta \mid v \models \alpha,\ v \not\models \beta}$$

$$\frac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$$

Proof by deduction: semantic arguments

To prove that a wff α is valid:

- Assume α is not valid, i.e., there is a interpretation v such that $v \not\models \alpha$
- Apply semantic arguments in the form of previous proof rules
- In the presence of multi-conclusion rules, proof evolves as a tree
  A proof tree branch is *closed* if it ends with $v \models \bot$, and is *open* otherwise
- A semantic argument is *finished* when no more proof rules are applicable
- It is a proof of the validity of α if every branch is closed
- Otherwise, each open branch describes an interpretation that falsifies α

Proof by deduction: semantic arguments

To prove that a wff α is valid:

- Assume α is not valid, i.e., there is a interpretation v such that $v \not\models \alpha$
- Apply semantic arguments in the form of previous proof rules
- In the presence of multi-conclusion rules, proof evolves as a tree
  A proof tree branch is *closed* if it ends with $v \models \bot$, and is *open* otherwise
- A semantic argument is *finished* when no more proof rules are applicable
- It is a proof of the validity of α if every branch is closed
- Otherwise, each open branch describes an interpretation that falsifies α

Proof by deduction: semantic arguments

To prove that a wff α is valid:

- Assume α is not valid, i.e., there is a interpretation v such that $v \not\models \alpha$
- Apply semantic arguments in the form of previous proof rules
- In the presence of multi-conclusion rules, proof evolves as a tree
  A proof tree branch is *closed* if it ends with $v \models \bot$, and is *open* otherwise
- A semantic argument is *finished* when no more proof rules are applicable
- It is a proof of the validity of α if every branch is closed
- Otherwise, each open branch describes an interpretation that falsifies α

Proof by deduction: semantic arguments

To prove that a wff α is valid:

- Assume α is not valid, i.e., there is a interpretation v such that $v \not\models \alpha$
- Apply semantic arguments in the form of previous proof rules
- In the presence of multi-conclusion rules, proof evolves as a tree
  A proof tree branch is *closed* if it ends with $v \models \bot$, and is *open* otherwise
- A semantic argument is *finished* when no more proof rules are applicable
- It is a proof of the validity of α if every branch is closed
- Otherwise, each open branch describes an interpretation that falsifies α

Proof by deduction: semantic arguments

To prove that a wff α is valid:

- Assume α is not valid, i.e., there is a interpretation v such that $v \not\models \alpha$
- Apply semantic arguments in the form of previous proof rules
- In the presence of multi-conclusion rules, proof evolves as a tree
  A proof tree branch is *closed* if it ends with $v \models \bot$, and is *open* otherwise
- A semantic argument is *finished* when no more proof rules are applicable
- It is a proof of the validity of α if every branch is closed
- Otherwise, each open branch describes an interpretation that falsifies α

Proof by deduction: semantic arguments

To prove that a wff α is valid:

- Assume α is not valid, i.e., there is a interpretation v such that $v \not\models \alpha$
- Apply semantic arguments in the form of previous proof rules
- In the presence of multi-conclusion rules, proof evolves as a tree
  A proof tree branch is *closed* if it ends with $v \models \bot$, and is *open* otherwise
- A semantic argument is *finished* when no more proof rules are applicable
- It is a proof of the validity of α if every branch is closed
- Otherwise, each open branch describes an interpretation that falsifies α

Proof by deduction: example

Prove $\alpha = p \land \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models p \land \neg q$ (assumption)
   - 1.1 $v \not\models p$ (by (d) on 1)
   - 1.2 $v \not\models \neg q$ (by (d) on 1)
     - 1.2.1 $v \models q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1:
  $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2:
  $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α
hence α is invalid

Proof by deduction: example

Prove $\alpha = p \land \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models p \land \neg q$ (assumption)
   - 1.1 $v \not\models p$ (by (d) on 1)
   - 1.2 $v \not\models \neg q$ (by (d) on 1)
     - 1.2.1 $v \models q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1:
  $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2:
  $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α
hence α is invalid

Proof by deduction: example

Prove $\alpha = p \wedge \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \wedge \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \wedge \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \vee \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \vee \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models p \wedge \neg q$ (assumption)

   1.1 $v \not\models p$ (by (d) on 1)

   1.2 $v \not\models \neg q$ (by (d) on 1)

   1.2.1 $v \models q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1: $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2: $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α
hence α is invalid

Proof by deduction: example

Prove $\alpha = p \land \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models p \land \neg q$ (assumption)
   - 1.1 $v \not\models p$ (by (d) on 1)
   - 1.2 $v \not\models \neg q$ (by (d) on 1)
     - 1.2.1 $v \models q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1: $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2: $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α
hence α is invalid

Proof by deduction: example

Prove $\alpha = p \wedge \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \vDash \neg\alpha}{v \nvDash \alpha}$

(b) $\dfrac{v \nvDash \neg\alpha}{v \vDash \alpha}$

(c) $\dfrac{v \vDash \alpha \wedge \beta}{v \vDash \alpha, v \vDash \beta}$

(d) $\dfrac{v \nvDash \alpha \wedge \beta}{v \nvDash \alpha \mid v \nvDash \beta}$

(e) $\dfrac{v \vDash \alpha \vee \beta}{v \vDash \alpha \mid v \vDash \beta}$

(f) $\dfrac{v \nvDash \alpha \vee \beta}{v \nvDash \alpha, v \nvDash \beta}$

(g) $\dfrac{v \vDash \alpha \Rightarrow \beta}{v \nvDash \alpha \mid v \vDash \beta}$

(h) $\dfrac{v \nvDash \alpha \Rightarrow \beta}{v \vDash \alpha, v \nvDash \beta}$

(i) $\dfrac{v \vDash \alpha \quad v \nvDash \alpha}{v \vDash \bot}$

(k) $\dfrac{v \vDash \alpha \Leftrightarrow \beta}{v \vDash \alpha, v \vDash \beta \mid v \nvDash \alpha, v \nvDash \beta}$

(j) $\dfrac{v \nvDash \alpha \Leftrightarrow \beta}{v \nvDash \alpha, v \vDash \beta \mid v \vDash \alpha, v \nvDash \beta}$

1. $v \nvDash p \wedge \neg q$ (assumption)
   - 1.1 $v \nvDash p$ (by (d) on 1)
   - 1.2 $v \nvDash \neg q$ (by (d) on 1)
     - 1.2.1 $v \vDash q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1: $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2: $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α

hence α is invalid

Proof by deduction: example

Prove $\alpha = p \wedge \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \wedge \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \wedge \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \vee \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \vee \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models p \wedge \neg q$ (assumption)

   1.1 $v \not\models p$ (by (d) on 1)

   1.2 $v \not\models \neg q$ (by (d) on 1)

   1.2.1 $v \models q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1:
  $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2:
  $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α
hence α is invalid

Proof by deduction: example

Prove $\alpha = p \land \neg q$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models p \land \neg q$ (assumption)

   1.1 $v \not\models p$ (by (d) on 1)

   1.2 $v \not\models \neg q$ (by (d) on 1)

      1.2.1 $v \models q$ (by (b) on 1.2)

Falsifying interpretations v:

- Branch 1.1:
  $\{p \mapsto \text{false}, q \mapsto \text{true/false}\}$
- Branch 1.2:
  $\{p \mapsto \text{true/false}, q \mapsto \text{true}\}$

there is at least a v that falsifies α
hence α is invalid

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \wedge \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \wedge \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \vee \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \vee \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \wedge (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \wedge \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \wedge \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \vee \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \vee \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \wedge (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \land (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \land (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \land (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \land (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \vDash \neg\alpha}{v \nvDash \alpha}$

(b) $\dfrac{v \nvDash \neg\alpha}{v \vDash \alpha}$

(c) $\dfrac{v \vDash \alpha \wedge \beta}{v \vDash \alpha, v \vDash \beta}$

(d) $\dfrac{v \nvDash \alpha \wedge \beta}{v \nvDash \alpha \mid v \nvDash \beta}$

(e) $\dfrac{v \vDash \alpha \vee \beta}{v \vDash \alpha \mid v \vDash \beta}$

(f) $\dfrac{v \nvDash \alpha \vee \beta}{v \nvDash \alpha, v \nvDash \beta}$

(g) $\dfrac{v \vDash \alpha \Rightarrow \beta}{v \nvDash \alpha \mid v \vDash \beta}$

(h) $\dfrac{v \nvDash \alpha \Rightarrow \beta}{v \vDash \alpha, v \nvDash \beta}$

(i) $\dfrac{v \vDash \alpha \quad v \nvDash \alpha}{v \vDash \bot}$

(k) $\dfrac{v \vDash \alpha \Leftrightarrow \beta}{v \vDash \alpha, v \vDash \beta \mid v \nvDash \alpha, v \nvDash \beta}$

(j) $\dfrac{v \nvDash \alpha \Leftrightarrow \beta}{v \nvDash \alpha, v \vDash \beta \mid v \vDash \alpha, v \nvDash \beta}$

1. $v \nvDash \alpha$ (assumption)
2. $v \vDash (p \Rightarrow q) \wedge (q \Rightarrow r)$ (by (h) on 1)
3. $v \nvDash p \Rightarrow r$ (by (h) on 1)
4. $v \vDash p$ (by (h) on 3)
5. $v \nvDash r$ (by (h) on 3)
6. $v \vDash p \Rightarrow q$ (by (c) on 2)
7. $v \vDash q \Rightarrow r$ (by (c) on 2)
8. $v \vDash q$ (by (l) on 4, 6)
9. $v \vDash r$ (by (l) on 7, 8)
10. $v \vDash \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \wedge \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \wedge \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \vee \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \vee \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \wedge (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \land (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \land \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \land \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \lor \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \lor \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \land (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Proof by deduction: example

Prove $\alpha = (p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is valid or find a falsifying interpretation

(a) $\dfrac{v \models \neg\alpha}{v \not\models \alpha}$

(b) $\dfrac{v \not\models \neg\alpha}{v \models \alpha}$

(c) $\dfrac{v \models \alpha \wedge \beta}{v \models \alpha, v \models \beta}$

(d) $\dfrac{v \not\models \alpha \wedge \beta}{v \not\models \alpha \mid v \not\models \beta}$

(e) $\dfrac{v \models \alpha \vee \beta}{v \models \alpha \mid v \models \beta}$

(f) $\dfrac{v \not\models \alpha \vee \beta}{v \not\models \alpha, v \not\models \beta}$

(g) $\dfrac{v \models \alpha \Rightarrow \beta}{v \not\models \alpha \mid v \models \beta}$

(h) $\dfrac{v \not\models \alpha \Rightarrow \beta}{v \models \alpha, v \not\models \beta}$

(i) $\dfrac{v \models \alpha \quad v \not\models \alpha}{v \models \bot}$

(k) $\dfrac{v \models \alpha \Leftrightarrow \beta}{v \models \alpha, v \models \beta \mid v \not\models \alpha, v \not\models \beta}$

(j) $\dfrac{v \not\models \alpha \Leftrightarrow \beta}{v \not\models \alpha, v \models \beta \mid v \models \alpha, v \not\models \beta}$

1. $v \not\models \alpha$ (assumption)
2. $v \models (p \Rightarrow q) \wedge (q \Rightarrow r)$ (by (h) on 1)
3. $v \not\models p \Rightarrow r$ (by (h) on 1)
4. $v \models p$ (by (h) on 3)
5. $v \not\models r$ (by (h) on 3)
6. $v \models p \Rightarrow q$ (by (c) on 2)
7. $v \models q \Rightarrow r$ (by (c) on 2)
8. $v \models q$ (by (l) on 4, 6)
9. $v \models r$ (by (l) on 7, 8)
10. $v \models \bot$ (by (i) on 5, 9)

Some useful tautologies

- Associative and Commutative laws
  - $\wedge$, $\vee$, and $\Leftrightarrow$
- Distributive laws
  - $\alpha \wedge (\beta \vee \gamma) \Leftrightarrow (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$
  - $\alpha \vee (\beta \wedge \gamma) \Leftrightarrow (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$
- Negation
  - $\neg\neg\alpha \Leftrightarrow \alpha$
  - $\neg(\alpha \Rightarrow \beta) \Leftrightarrow (\alpha \wedge \neg\beta)$
  - $\neg(\alpha \Leftrightarrow \beta) \Leftrightarrow (\alpha \wedge \neg\beta) \vee (\neg\alpha \wedge \beta)$
- De Morgan's laws
  - $\neg(\alpha \wedge \beta) \Leftrightarrow (\neg\alpha \vee \neg\beta)$
  - $\neg(\alpha \vee \beta) \Leftrightarrow (\neg\alpha \wedge \neg\beta)$
- Implication
  - $(\alpha \Rightarrow \beta) \Leftrightarrow (\neg\alpha \vee \beta)$
- Excluded Middle
  - $\alpha \vee \neg\alpha$
- Contradiction
  - $\neg(\alpha \wedge \neg\alpha)$
- Contraposition
  - $(\alpha \Rightarrow \beta) \Leftrightarrow (\neg\beta \Rightarrow \neg\alpha)$
- Exportation
  - $((\alpha \wedge \beta) \Rightarrow \gamma) \Leftrightarrow (\alpha \Rightarrow (\beta \Rightarrow \gamma))$

Some useful tautologies

- Associative and Commutative laws
  - $\wedge$, $\vee$, and $\Leftrightarrow$
- Distributive laws
  - $\alpha \wedge (\beta \vee \gamma) \Leftrightarrow (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$
  - $\alpha \vee (\beta \wedge \gamma) \Leftrightarrow (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$
- Negati
  - $\neg\neg\alpha \Leftrightarrow \alpha$
  - $\neg(\alpha \Rightarrow \beta) \Leftrightarrow (\alpha \wedge \neg\beta)$
  - $\neg(\alpha \Leftrightarrow \beta) \Leftrightarrow (\alpha \wedge \neg\beta) \vee (\neg\alpha \wedge \beta)$
- De Morgan's laws
  - $\neg(\alpha \wedge \beta) \Leftrightarrow (\neg\alpha \vee \neg\beta)$
  - $\neg(\alpha \vee \beta) \Leftrightarrow (\neg\alpha \wedge \neg\beta)$
- Implication
  - $(\alpha \Rightarrow \beta) \Leftrightarrow (\neg\alpha \vee \beta)$
- Excluded Middle
  - $\alpha \vee \neg\alpha$
- Contraposition
  - $(\alpha \Rightarrow \beta) \Leftrightarrow (\neg\beta \Rightarrow \neg\alpha)$
- Exportation
  - $((\alpha \wedge \beta) \Rightarrow \gamma) \Leftrightarrow (\alpha \Rightarrow (\beta \Rightarrow \gamma))$

These tautologies can be proven with semantic arguments

Semantic arguments for satisfiability

The previous proof system was used to prove a formula is valid

It can also be used to prove that a formula α is unsatisfiable:

1. Again by contradiction, start with the assertion $v \models \alpha$
2. Try to derive a proof tree T whose branches are all closed

Such a tree proves that α is unsatisfiable

If T has an open branch B where no (more) rules apply
then α is satisfiable with an interpretation v constructible from B

Semantic arguments for satisfiability

The previous proof system was used to prove a formula is valid

It can also be used to prove that a formula α is unsatisfiable:

1. Again by contradiction, start with the assertion $v \models \alpha$
2. Try to derive a proof tree T whose branches are all closed

Such a tree proves that α is unsatisfiable

If T has an open branch B where no (more) rules apply
then α is satisfiable with an interpretation v constructible from B

Semantic arguments for satisfiability

The previous proof system was used to prove a formula is valid

It can also be used to prove that a formula α is unsatisfiable:

1. Again by contradiction, start with the assertion $v \models \alpha$
2. Try to derive a proof tree T whose branches are all closed

Such a tree proves that α is unsatisfiable

If T has an open branch B where no (more) rules apply
then α is satisfiable with an interpretation v constructible from B

Semantic arguments for satisfiability

The previous proof system was used to prove a formula is valid

It can also be used to prove that a formula α is unsatisfiable:

1. Again by contradiction, start with the assertion $v \models \alpha$
2. Try to derive a proof tree T whose branches are all closed

Such a tree proves that α is unsatisfiable

If T has an open branch B where no (more) rules apply
then α is satisfiable with an interpretation v constructible from B

Semantic arguments for satisfiability

The previous proof system was used to prove a formula is valid

It can also be used to prove that a formula α is unsatisfiable:

1. Again by contradiction, start with the assertion $v \models \alpha$
2. Try to derive a proof tree T whose branches are all closed

Such a tree proves that α is unsatisfiable

If T has an open branch B where no (more) rules apply
then α is satisfiable with an interpretation v constructible from B

Deductive systems

A *deductive system* $\mathscr{D}$ is a proof system equipped with a distinguished set of tautologies (*axioms*)

A *proof* in $\mathscr{D}$ for a wff α_n is a sequence of formulas $S = (\alpha_1, \ldots, \alpha_n)$ where each α_i is

- either an axiom
- or the result of an application of a rule of $\mathscr{D}$ to previous formulas in S

In that case, α_n is *provable* or a *theorem* in $\mathscr{D}$, written as $\vdash \alpha_n$

For $U \subseteq \mathcal{W}$, we write $U \vdash \alpha$ to denote that α can be proved in $\mathscr{D}$ from the axioms and the formulas in U

We call $U \vdash \alpha$ a *sequent*

Deductive systems

A *deductive system* $\mathscr{D}$ is a proof system equipped with a distinguished set of tautologies (*axioms*)

A *proof* in $\mathscr{D}$ for a wff α_n is a sequence of formulas $S = (\alpha_1, \ldots, \alpha_n)$ where each α_i is

- either an axiom
- or the result of an application of a rule of $\mathscr{D}$ to previous formulas in S

In that case, α_n is *provable* or a *theorem* in $\mathscr{D}$, written as $\vdash \alpha_i$

For $U \subseteq \mathcal{W}$, we write $U \vdash \alpha$ to denote that α can be proved in $\mathscr{D}$ from the axioms and the formulas in U

We call $U \vdash \alpha$ a *sequent*

Deductive systems

A *deductive system* $\mathscr{D}$ is a proof system equipped with a distinguished set of tautologies (*axioms*)

A *proof* in $\mathscr{D}$ for a wff α_n is a sequence of formulas $S = (\alpha_1, \ldots, \alpha_n)$ where each α_i is

- either an axiom
- or the result of an application of a rule of $\mathscr{D}$ to previous formulas in S

In that case, α_n is *provable* or a *theorem* in $\mathscr{D}$, written as $\vdash \alpha_i$

For $U \subseteq \mathcal{W}$, we write $U \vdash \alpha$ to denote that α can be proved in $\mathscr{D}$ from the axioms and the formulas in U

We call $U \vdash \alpha$ a *sequent*

Deductive systems

A *deductive system* $\mathscr{D}$ is a proof system equipped with a distinguished set of tautologies (*axioms*)

A *proof* in $\mathscr{D}$ for a wff α_n is a sequence of formulas $S = (\alpha_1, \ldots, \alpha_n)$ where each α_i is

- either an axiom
- or the result of an application of a rule of $\mathscr{D}$ to previous formulas in S

In that case, α_n is *provable* or a *theorem* in $\mathscr{D}$, written as $\vdash \alpha_i$

For $U \subseteq \mathcal{W}$, we write $U \vdash \alpha$ to denote that α can be proved in $\mathscr{D}$ from the axioms and the formulas in U

We call $U \vdash \alpha$ a *sequent*

Deductive systems

Important properties of a deductive system w.r.t. a logic's semantics:

Deductive systems

Important properties of a deductive system w.r.t. a logic's semantics:

- Consistency: for all α, at most one of α and $\neg\alpha$ is provable
- Soundness: If $\vdash \alpha$, then $\vDash \alpha$
- Completeness: If $\vDash \alpha$, then $\vdash \alpha$

Deductive systems

Important properties of a deductive system w.r.t. a logic's semantics:

- Consistency: for all α, at most one of α and $\neg\alpha$ is provable
- Soundness: If $\vdash \alpha$, then $\models \alpha$
- Completeness: If $\models \alpha$, then $\vdash \alpha$

Deductive systems

Important properties of a deductive system w.r.t. a logic's semantics:

- Consistency: for all α, at most one of α and $\neg\alpha$ is provable
- Soundness: If $\vdash \alpha$, then $\models \alpha$
- Completeness: If $\models \alpha$, then $\vdash \alpha$

Hilbert System $\mathscr{H}_2$

A consistent, sound and complete deductive system for propositional logic

Axiom schemas (α, β, γ are arbitrary wffs):

A1: $\vdash \alpha \Rightarrow (\beta \Rightarrow \alpha)$

A2: $\vdash (\alpha \Rightarrow (\beta \Rightarrow \gamma)) \Rightarrow ((\alpha \Rightarrow \beta) \Rightarrow (\alpha \Rightarrow \gamma))$

A3: $\vdash (\neg\beta \Rightarrow \neg\alpha) \Rightarrow (\alpha \Rightarrow \beta)$

Rules

$$\frac{\vdash \alpha \qquad \vdash \alpha \Rightarrow \beta}{\vdash \beta}$$ (modus ponens)

Hilbert System $\mathscr{H}_2$

A consistent, sound and complete deductive system for propositional logic

Axiom schemas (α, β, γ are arbitrary wffs):

A1: $\vdash \alpha \Rightarrow (\beta \Rightarrow \alpha)$

A2: $\vdash (\alpha \Rightarrow (\beta \Rightarrow \gamma)) \Rightarrow ((\alpha \Rightarrow \beta) \Rightarrow (\alpha \Rightarrow \gamma))$

A3: $\vdash (\neg\beta \Rightarrow \neg\alpha) \Rightarrow (\alpha \Rightarrow \beta)$

Rules

$$\frac{\vdash \alpha \qquad \vdash \alpha \Rightarrow \beta}{\vdash \beta} \quad \text{(modus ponens)}$$

Hilbert System $\mathscr{H}_2$

A consistent, sound and complete deductive system for propositional logic

Axiom schemas (α, β, γ are arbitrary wffs):

A1: $\vdash \alpha \Rightarrow (\beta \Rightarrow \alpha)$

A2: $\vdash (\alpha \Rightarrow (\beta \Rightarrow \gamma)) \Rightarrow ((\alpha \Rightarrow \beta) \Rightarrow (\alpha \Rightarrow \gamma))$

A3: $\vdash (\neg\beta \Rightarrow \neg\alpha) \Rightarrow (\alpha \Rightarrow \beta)$

Rules

$$\frac{\vdash \alpha \qquad \vdash \alpha \Rightarrow \beta}{\vdash \beta} \text{ (modus ponens)}$$

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Example: Prove $\varphi \Rightarrow \varphi$

1. $\vdash (\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ (by A2)
2. $\vdash \varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ (by A1)
3. $\vdash (\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ (by MP 1, 2)
4. $\vdash \varphi \Rightarrow (\varphi \Rightarrow \varphi)$ (by A1)
5. $\vdash \varphi \Rightarrow \varphi$ (by MP 3, 4)

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Example: Prove $\varphi \Rightarrow \varphi$

1. $\vdash (\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ (by A2)
2. $\vdash \varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ (by A1)
3. $\vdash (\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ (by MP 1, 2)
4. $\vdash \varphi \Rightarrow (\varphi \Rightarrow \varphi)$ (by A1)
5. $\vdash \varphi \Rightarrow \varphi$ (by MP 3, 4)

A2: $\vdash (\alpha \Rightarrow (\beta \Rightarrow \gamma)) \Rightarrow ((\alpha \Rightarrow \beta) \Rightarrow (\alpha \Rightarrow \gamma))$

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Example: Prove $\varphi \Rightarrow \varphi$

1. $\vdash (\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ (by A2)
2. $\vdash \varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ (by A1)
3. $\vdash (\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ (by MP 1, 2)
4. $\vdash \varphi \Rightarrow (\varphi \Rightarrow \varphi)$ (by A1)
5. $\vdash \varphi \Rightarrow \varphi$ (by MP 3, 4)

A1: $\vdash \alpha \Rightarrow (\beta \Rightarrow \alpha)$

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Example: Prove $\varphi \Rightarrow \varphi$

1. $\vdash (\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ (by A2)
2. $\vdash \varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ (by A1)
3. $\vdash (\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ (by MP 1, 2)
4. $\vdash \varphi \Rightarrow (\varphi \Rightarrow \varphi)$ (by A1)
5. $\vdash \varphi \Rightarrow \varphi$ (by MP 3, 4)

$$\frac{\vdash \alpha \qquad \vdash \alpha \Rightarrow \beta}{\vdash \beta}$$ (modus ponens)

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Example: Prove $\varphi \Rightarrow \varphi$

1. $\vdash (\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ (by A2)
2. $\vdash \varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ (by A1)
3. $\vdash (\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ (by MP 1, 2)
4. $\vdash \varphi \Rightarrow (\varphi \Rightarrow \varphi)$ (by A1)
5. $\vdash \varphi \Rightarrow \varphi$ (by MP 3, 4)

A1: $\vdash \alpha \Rightarrow (\beta \Rightarrow \alpha)$

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Example: Prove $\varphi \Rightarrow \varphi$

1. $\vdash (\varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)) \Rightarrow ((\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi))$ (by A2)
2. $\vdash \varphi \Rightarrow ((\varphi \Rightarrow \varphi) \Rightarrow \varphi)$ (by A1)
3. $\vdash (\varphi \Rightarrow (\varphi \Rightarrow \varphi)) \Rightarrow (\varphi \Rightarrow \varphi)$ (by MP 1, 2)
4. $\vdash \varphi \Rightarrow (\varphi \Rightarrow \varphi)$ (by A1)
5. $\vdash \varphi \Rightarrow \varphi$ (by MP 3, 4)

$$\frac{\vdash \alpha \qquad \vdash \alpha \Rightarrow \beta}{\vdash \beta} \text{ (modus ponens)}$$

Proofs in $\mathscr{H}_2$

Proofs can be complicated, even for trivial formulas (or formula schemas)

Solution:

Introduce *derived* proof rules, additional rules whose conclusion can be proved from their premises using no derived proof rules

Derived Rules in $\mathscr{H}_2$

$$\frac{}{U \cup \{\alpha\} \vdash \alpha} \text{ (assumption)}$$

$$\frac{U \cup \{\alpha\} \vdash \beta}{U \vdash \alpha \Rightarrow \beta} \text{ (deduction)}$$

$$\frac{U \vdash \neg\beta \Rightarrow \neg\alpha}{U \vdash \alpha \Rightarrow \beta} \text{ (contrapositive)}$$

$$\frac{U \vdash \neg\neg\alpha}{U \vdash \alpha} \text{ (double negation 1)}$$

$$\frac{U \vdash \alpha \Rightarrow \beta \qquad U \vdash \beta \Rightarrow \gamma}{U \vdash \alpha \Rightarrow \gamma} \text{ (transitivity)}$$

$$\frac{U \vdash \alpha}{U \vdash \neg\neg\alpha} \text{ (double negation 2)}$$

$$\frac{U \vdash \alpha \Rightarrow (\beta \Rightarrow \gamma)}{U \vdash \beta \Rightarrow (\alpha \Rightarrow \gamma)} \text{ (exchange of antecedent)}$$

$$\frac{U \vdash \neg\alpha \Rightarrow \bot}{U \vdash \alpha} \text{ (reductio ad absurdum)}$$

Using derived rules in $\mathscr{H}_2$

With the deduction rule, the proof of $\alpha \Rightarrow \alpha$ becomes trivial

1. $\{\alpha\} \vdash \alpha$ (by assumption)
2. $\vdash \alpha \Rightarrow \alpha$ (by deduction on 1)

This is because we front-load the proof burden in proving that the assumption and the deduction rule are derived rules

Using derived rules in $\mathscr{H}_2$

With the deduction rule, the proof of $\alpha \Rightarrow \alpha$ becomes trivial

1. $\{\, \alpha \,\} \vdash \alpha$ (by assumption)
2. $\vdash \alpha \Rightarrow \alpha$ (by deduction on 1)

This is because we front-load the proof burden in proving that the assumption and the deduction rule are derived rules

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{}{U \vdash \alpha \Rightarrow (\beta \Rightarrow \alpha)}\ \text{(A1)}$$

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{}{U \cup \{\alpha\} \vdash \alpha} \text{ (assumption)}$$

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{U \vdash \alpha \qquad U \vdash \alpha \Rightarrow \beta}{U \vdash \beta} \text{ (modus ponens)}$$

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{}{U \vdash (\neg\beta \Rightarrow \neg\alpha) \Rightarrow (\alpha \Rightarrow \beta)}\ \text{(A3)}$$

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{U \vdash \alpha \qquad U \vdash \alpha \Rightarrow \beta}{U \vdash \beta} \text{ (modus ponens)}$$

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{U \cup \{\alpha\} \vdash \beta}{U \vdash \alpha \Rightarrow \beta} \ (\text{deduction})$$

Using derived rules in $\mathscr{H}_2$

Example 1: prove $\varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

1. $\{\neg\varphi\} \vdash \neg\varphi \Rightarrow (\neg\psi \Rightarrow \neg\varphi)$ (A1)
2. $\{\neg\varphi\} \vdash \neg\varphi$ (assumption)
3. $\{\neg\varphi\} \vdash \neg\psi \Rightarrow \neg\varphi$ (MP 1, 2)
4. $\{\neg\varphi\} \vdash (\neg\psi \Rightarrow \neg\varphi) \Rightarrow (\varphi \Rightarrow \psi)$ (A3)
5. $\{\neg\varphi\} \vdash \varphi \Rightarrow \psi$ (MP 3, 4)
6. $\vdash \neg\varphi \Rightarrow (\varphi \Rightarrow \psi)$ (deduction)
7. $\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$ (exchange of antecedent)

$$\frac{U \vdash \alpha \Rightarrow (\beta \Rightarrow \gamma)}{U \vdash \beta \Rightarrow (\alpha \Rightarrow \gamma)}$$ (exchange of antecedent)

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{}{U \cup \{\alpha\} \vdash \alpha} \text{ (assumption)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \vdash \neg\neg\alpha}{U \vdash \alpha} \quad \text{(double negation 1)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{}{U \cup \{\alpha\} \vdash \alpha} \text{ (assumption)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \vdash \alpha \qquad U \vdash \alpha \Rightarrow \beta}{U \vdash \beta} \text{ (modus ponens)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$\vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \psi)$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \vdash \alpha \qquad U \vdash \alpha \Rightarrow \beta}{U \vdash \beta} \text{ (modus ponens)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \vdash \alpha \qquad U \vdash \alpha \Rightarrow \beta}{U \vdash \beta} \text{ (modus ponens)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \cup \{\alpha\} \vdash \beta}{U \vdash \alpha \Rightarrow \beta} \text{ (deduction)}$$

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \vdash \neg\alpha \Rightarrow \bot}{U \vdash \alpha}$$ (reductio ad absurdum)

Soundness of rules in $\mathscr{H}_2$

Example 2: prove $(\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$

1. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\neg\varphi$ (assumption)
2. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi$ (double negation 1)
3. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow \neg\varphi$ (assumption)
4. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi$ (MP 2, 3)
5. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \varphi \Rightarrow (\neg\varphi \Rightarrow \bot)$ (Ex. 1)
6. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \neg\varphi \Rightarrow \bot$ (MP 2, 5)
7. $\{\varphi \Rightarrow \neg\varphi, \neg\neg\varphi\} \vdash \bot$ (MP 4, 6)
8. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\neg\varphi \Rightarrow \bot$ (deduction 7)
9. $\{\varphi \Rightarrow \neg\varphi\} \vdash \neg\varphi$ (reductio ad absurdum 8)
10. $\vdash (\varphi \Rightarrow \neg\varphi) \Rightarrow \neg\varphi$ (deduction 9)

$$\frac{U \cup \{\alpha\} \vdash \beta}{U \vdash \alpha \Rightarrow \beta} \text{ (deduction)}$$

Soundness of rules in $\mathscr{H}_2$

A proof rule

$$\frac{U_1 \vdash \alpha_1 \quad \cdots \quad U_n \vdash \alpha_n}{V \vdash \beta}$$

is *sound* if $V \vDash \beta$ whenever $U_1 \vDash \alpha_1, \ldots, U_n \vDash \alpha_n$

Theorem: Axioms 1–3, modus ponens, and all the derived rules of $\mathscr{H}_2$ are sound

Soundness of rules in $\mathscr{H}_2$

A proof rule

$$\frac{U_1 \vdash \alpha_1 \quad \cdots \quad U_n \vdash \alpha_n}{V \vdash \beta}$$

is *sound* if $V \models \beta$ whenever $U_1 \models \alpha_1, \ldots, U_n \models \alpha_n$

Theorem: Axioms 1–3, modus ponens, and all the derived rules of $\mathscr{H}_2$ are sound

All rules of $\mathscr{H}_2$ are sound

$$\frac{\vdash \alpha \qquad \vdash \alpha \Rightarrow \beta}{\vdash \beta} \text{ (modus ponens)}$$

$$\frac{}{U \cup \{\alpha\} \vdash \alpha} \text{ (assumption)}$$

$$\frac{U \cup \{\alpha\} \vdash \beta}{U \vdash \alpha \Rightarrow \beta} \text{ (deduction)}$$

$$\frac{U \vdash \neg\beta \Rightarrow \neg\alpha}{U \vdash \alpha \Rightarrow \beta} \text{ (contrapositive)}$$

$$\frac{U \vdash \neg\neg\alpha}{U \vdash \alpha} \text{ (double negation 1)}$$

$$\frac{U \vdash \alpha \Rightarrow \beta \qquad U \vdash \beta \Rightarrow \gamma}{U \vdash \alpha \Rightarrow \gamma} \text{ (transitivity)}$$

$$\frac{U \vdash \alpha}{U \vdash \neg\neg\alpha} \text{ (double negation 2)}$$

$$\frac{U \vdash \alpha \Rightarrow (\beta \Rightarrow \gamma)}{U \vdash \beta \Rightarrow (\alpha \Rightarrow \gamma)} \text{ (exchange of antecedent)}$$

$$\frac{U \vdash \neg\alpha \Rightarrow \bot}{U \vdash \alpha} \text{ (reductio ad absurdum)}$$

Alternative proof systems

Another way to define a proof system is to

- include more logical connectives and
- have lots of proof rules

Proofs can be simpler to (manually) carry out in such a system

However, it becomes harder to prove properties about the proof system

Either way, Hilbert-style proof systems are difficult to automate

We will focus on proof systems more similar to semantic arguments

Alternative proof systems

Another way to define a proof system is to

- include more logical connectives and
- have lots of proof rules

Proofs can be simpler to (manually) carry out in such a system

However, it becomes harder to prove properties about the proof system

Either way, Hilbert-style proof systems are difficult to automate

We will focus on proof systems more similar to semantic arguments

Alternative proof systems

Another way to define a proof system is to

- include more logical connectives and
- have lots of proof rules

Proofs can be simpler to (manually) carry out in such a system

However, it becomes harder to prove properties about the proof system

Either way, Hilbert-style proof systems are difficult to automate

We will focus on proof systems more similar to semantic arguments

Alternative proof systems

Another way to define a proof system is to

- include more logical connectives and
- have lots of proof rules

Proofs can be simpler to (manually) carry out in such a system

However, it becomes harder to prove properties about the proof system

Either way, Hilbert-style proof systems are difficult to automate

We will focus on proof systems more similar to semantic arguments

Alternative proof systems

Another way to define a proof system is to

- include more logical connectives and
- have lots of proof rules

Proofs can be simpler to (manually) carry out in such a system

However, it becomes harder to prove properties about the proof system

Either way, Hilbert-style proof systems are difficult to automate

We will focus on proof systems more similar to semantic arguments