LOGICS FOR COMPUTER SCIENCE: Classical and Non-Classical

Anita Wasilewska

Chapter 8
Classical Predicate Semantics and Proof Systems

CHAPTER 8 SLIDES

Slides Set 3

Chapter 8
Classical Predicate Semantics and Proof Systems

Slides Set 3

PART 3: Predicate Tautologies, Equational Laws of Quantifiers

PART 4: Proof Systems: Soundness and Completeness

Chapter 8
Classical Predicate Semantics and Proof Systems

Slides Set 3

PART 3: Predicate Tautologies, Equational Laws of Quantifiers

Predicate Tautologies

Predicate Tautologies

We have already proved the **basic** predicate tautology

$$\models \ (\forall x\, A(x) \Rightarrow \exists x\, A(x))$$

We **prove** now other three **basic** tautologies called

Dictum de Omni

For any formula $A(x)$ of $\mathcal{L}$,

$$\models \ (\forall x\, A(x) \Rightarrow A(t)), \qquad \models \ (\forall x\, A(x) \Rightarrow A(x))$$

$$\models \ (A(t) \Rightarrow \exists x\, A(x))$$

where t is a term, A(t) is a result of substitution of t for all free occurrences of x in $A(x)$, and t is **free for** x in A(x), i.e. **no** occurrence of a variable in t becomes a **bound** occurrence in A(t)

Proof of Dictum de Omni

Proof of

$$\models \ (\forall x\, A(x) \Rightarrow A(t)), \qquad \models \ (\forall x\, A(x) \Rightarrow A(x))$$

is constructed in a sequence of the following steps

We leave details to complete as an exercise

S1

Consider a structure $\mathbf{M} = [U, I]$ and $s : VAR \longrightarrow U$

Let $t,\ u$ be two terms

Denote by t' a result of **replacing** in t all occurrences of a variable x by the term u, i.e.

$$t' = t(x/u)$$

Let s' results from s by **replacing** $s(x)$ by $s_I(u)$

We prove by induction over the length of t that

$$s_I(t(x/u)) = s_I(t') = s'_I(u)$$

Proof of Dictum de Omni

S2

Let t be **free for** x in A(x)

$A(t)$ is a results from $A(x)$ by replacing t for all free occurrences of x in $A(x)$, i.e.

$$A(t) = A(x/t)$$

Let

$$s : VAR \longrightarrow U$$

and s' be obtained from s by replacing $s(x)$ by $s_I(u)$

We use

$$s_I(t(x/u)) = s_I(t') = s'_I(u)$$

and induction on the number of connectives and quantifiers in $A(x)$ and prove

$$(\mathbf{M}, s) \models A(x/t) \quad \text{if and only if} \quad (\mathbf{M}, s') \models A(x)$$

Proof of Dictum de Omni

S3

Directly from satisfaction definition and

$$(\mathbf{M}, s) \models A(x/t) \quad \text{if and only if} \quad (\mathbf{M}, s') \models A(x)$$

we get that for any $\mathbf{M} = [U, I]$ and any $s : VAR \longrightarrow U$,

$$\text{if } (\mathbf{M}, s) \models \forall x A(x), \text{ then } (\mathbf{M}, s) \models A(t)$$

This proves

$$\models (\forall x\, A(x) \Rightarrow A(t))$$

Observe that obviously a term x is **free for** x in A(x), so we also get as a **particular** case of $t = x$ that

$$\models (\forall x\, A(x) \Rightarrow A(x))$$

Dictum de Omni Restrictions

Proof of

$$\models \ (A(t) \Rightarrow \exists x\, A(x))$$

is included in detail in Section 3

Remark

The **restrictions** on terms in Dictum de Omni tautologies are **essential**

Here is a simple example explaining why they are needed in

$$\models \ (\forall x\, A(x) \Rightarrow A(t)), \qquad \models \ (\forall x\, A(x) \Rightarrow A(x))$$

Let $A(x)$ be a formula

$$\neg \forall y\, P(x, y) \quad \text{for} \quad P \in \mathbf{P}$$

Notice that a **term** $t = y$ is **not free for y** in A(x)

Dictum de Omni Restrictions

Consider the first formula in **Dictum de Omni** for $A(x) = \neg\forall y\, P(x, y)$ and term $t = y$

$$(\forall x \neg\forall y\, P(x, y) \Rightarrow \neg\forall y\, P(y, y))$$

Take

$$\mathbf{M} = [N, I] \quad \text{for } \mathbf{I} \text{ such that } P_I := =$$

Obviously,

$$\mathbf{M} \models \forall x \neg\forall y\, P(x, y)$$

as

$$\forall m\, \neg\forall n (m = n)$$

is a **true** mathematical statement in the set N of natural numbers

Dictum de Omni Restrictions

$$\mathbf{M} \not\models \neg\forall y\ P(y, y)$$

as

$$\neg\forall n\ (n = n)$$

is a **false** statement for $n \in N$

The second **Dictum de Omni** formula is a particular case of the first

We have proved that without the restrictions on terms

$$\not\models\ (\forall x\ A(x) \Rightarrow A(t)) \quad \text{and} \quad \not\models\ (\forall x\ A(x) \Rightarrow A(x))$$

The example for $\models\ (A(t) \Rightarrow \exists x\ A(x))$ is similar

"t free for x in $A(x)$"

Here are some **useful** and easy to prove **properties** of the notion " term t **free for** x in $A(x)$ "

Properties

For any formula $A \in \mathcal{F}$ and any term $t \in \mathbf{T}$ the following properties hold

P1. Closed term t, i.e. term with no variables is free for any variable x in A

P2. Term t is free for any variable in A if none of the variables in t is bound in A

P3. Term $t = x$ is free for x in any formula A

P4. Any term is free for x in A if A contains no free occurrences of x

Predicate Tautologies

Here are some more **important** predicate **tautologies**

For any formulas $A(x), B(x), A, B$ of $\mathcal{L}$, where the formulas A, B **do not** contain any free occurrences of x the following holds

Generalization

$$\models ((B \Rightarrow A(x)) \Rightarrow (B \Rightarrow \forall x\, A(x)))$$

$$\models ((B(x) \Rightarrow A) \Rightarrow (\exists x B(x) \Rightarrow A))$$

Distributivity 1

$$\models (\forall x(A \Rightarrow B(x)) \Rightarrow (A \Rightarrow \forall x\, B(x)))$$

$$\models \forall x(A(x) \Rightarrow B) \Rightarrow (\exists x A(x) \Rightarrow B)$$

$$\models \exists x(A(x) \Rightarrow B) \Rightarrow (\forall x A(x) \Rightarrow B)$$

Restrictions

The **restrictions** that the formulas A, B **do not** contain any free occurrences of x is **essential** for both Generalization and Distributivity 1 tautologies

Here is a simple **example** explaining why they are needed

The relaxation of the **restrictions** would lead to the following disaster

Let A and B be both the same **atomic** formula P(x)

Thus x is **free** in A and we have the following instance of the first .Distributivity 1 tautology

$$.(\forall x(P(x) \Rightarrow P(x)) \Rightarrow (P(x) \Rightarrow \forall x\, P(x)))$$

Restrictions

Take

$$\mathbf{M} = [N, I\,] \quad \text{for } \mathbf{I} \text{ such that } P_I = ODD$$

where $ODD \subseteq N$ is the set of odd numbers

Let $s : VAR \longrightarrow N$

By definition of the interpretation i,

$$s_I(x) \in P_I \quad \text{if and only if} \quad s_I(x) \in ODD$$

Then obviously

$$(\mathbf{M}, s) \not\models \forall x\, P(x)$$

and $\mathbf{M} = [N, I\,]$ is a **counter model** for

$$(\forall x(P(x) \Rightarrow P(x)) \Rightarrow (P(x) \Rightarrow \forall x\, P(x)))$$

as

$$\models \forall x(P(x) \Rightarrow P(x))$$

The examples for restrictions on other tautologies are similar.

Predicate Tautologies

Distributivity 2

For any formulas $A(x), B(x)$ of $\mathcal{L}$

$$\models (\exists x\,(A(x) \cap B(x)) \Rightarrow (\exists x A(x) \cap \exists x B(x)))$$

$$\models ((\forall x A(x) \cup \forall x B(x)) \Rightarrow \forall x\,(A(x) \cup B(x)))$$

$$\models (\forall x(A(x) \Rightarrow B(x)) \Rightarrow (\forall x A(x) \Rightarrow \forall x B(x)))$$

The converse implications to the above **are not** predicate tautologies

The **counter models** are provided in the Section 3

De Morgan Laws

For any formulas $A(x), B(x)$ of $\mathcal{L}$,

$$\models (\neg\forall x A(x) \Rightarrow \exists x \neg A(x))$$

$$\models (\neg\exists x A(x) \Rightarrow \forall x \neg A(x))$$

$$\models (\exists x \neg A(x) \Rightarrow \neg\forall x A(x))$$

$$\models (\neg\exists x A(x) \Rightarrow \forall x \neg A(x))$$

We prove the first law as an example

The proofs of all other laws are similar

De Morgan Laws

Proof of

$$\models (\neg \forall x A(x) \Rightarrow \exists x \neg A(x))$$

We carry the proof by **contradiction**

Assume that

$$\not\models \models (\neg \forall x A(x) \Rightarrow \exists x \neg A(x))$$

By definition, there is

$$\mathbf{M} = [U, I] \quad \text{and} \quad s : VAR \longrightarrow U$$

such that

$$(\mathbf{M}, s) \models \neg \forall x A(x)) \quad \text{and} \quad (\mathbf{M}, s) \not\models \exists x \neg A(x)$$

De Morgan Laws

Consider

$$(\mathbf{M}, s) \models \neg \forall x A(x)$$

By satisfaction definition

$$(\mathbf{M}, s) \not\models \forall x A(x)$$

This holds only if for **all** s', such that s, s' agree on all variables except on x,

$$(\mathbf{M}, s') \not\models A(x)$$

De Morgan Laws

Consider now

$$(\mathbf{M}, s) \not\models \exists x \neg A(x)$$

This holds only if **there is no** s', such that

$$(\mathbf{M}, s') \models \neg A(x)$$

i.e. there **is no** s', such that $(\mathbf{M}, s') \not\models A(x)$

This means that **for all** s',

$$(\mathbf{M}, s') \models A(x)$$

Contradiction with already proved

$$(\mathbf{M}, s') \not\models A(x)$$

This **ends** the proof

Quantifiers Alternations

For any formula $A(x, y)$ of $\mathcal{L}$,

$$\models (\exists x \forall y A(x, y) \Rightarrow \forall y \exists x A(x, y))$$

The converse implication

$$(\forall y \exists x A(x, y) \Rightarrow \exists x \forall y A(x, y))$$

is not a predicate **tautology**

Here is a proof

Take as $A(x, y)$ an atomic formula $R(x, y)$

Consider the **instance** formula

$$(\forall y \exists x R(x, y) \Rightarrow \exists x \forall y R(x, y))$$

Quantifiers Alternations

We construct now a counter model for the instance formula

$$(\forall y \exists x R(x, y) \Rightarrow \exists x \forall y R(x, y))$$

Take a structure

$$\mathbf{M} = [R, I]$$

where R is the set of real numbers and $R_I :<$

The instance formula becomes a mathematical statement

$$(\forall y \exists x (x < y) \Rightarrow \exists x \forall y (x < y))$$

that obviously **false** in the set of real numbers

We proved

$$\not\models (\forall y \exists x A(x, y) \Rightarrow \exists x \forall y A(x, y))$$

Equational Laws of Quantifiers

Logical Equivalence

The most frequently used **laws of quantifiers** have a form of a logical equivalence, symbolically written as $\equiv$

Logical equivalence $\equiv$ **is not** a new logical connective but just a very useful **symbol**

Logical equivalence $\equiv$ has the same properties as the mathematical equality $=$ and can be used in a similar way as we use the equality

Note that we use the same equivalence symbol $\equiv$ and the tautology symbol $\models$ for **propositional** and **predicate** languages when there is no confusion

Logical Equivalence

We define formally the **logical equivalence** $\equiv$ as follows.

Definition of Logical Equivalence

For any formulas A, B of the **predicate** language $\mathcal{L}$,

$$A \equiv B \text{ if and only if } \models (A \Rightarrow B) \text{ and } \models (B \Rightarrow A)$$

Remark that the predicate language $\mathcal{L}$ we defined the **semantics** for does not include the equivalence connective $\Leftrightarrow$. If it does we **extend** the satisfaction definition in a natural way and adopt the following, natural definition

Definition

For any formulas $A, B \in \mathcal{F}$ of the **predicate language** $\mathcal{L}$ with the equivalence connective $\Leftrightarrow$

$$A \equiv B \quad \text{if and only if} \quad \models (A \Leftrightarrow B)$$

Logical Equivalence Theorems

The **basic** theorems establishing relationship between propositional and some predicate **tautologies** are as follows

Tautologies Theorem

If a formula A is a propositional tautology,

then by **substituting** for propositional variables in A any formula of the **predicate** language $\mathcal{L}$ we obtain a formula which is a predicate tautology

Equivalences Theorem

Given propositional formulas A, B

If $A \equiv B$ is a propositional **equivalence**, and

A', B' are formulas of the predicate language L obtained by a **substitution** of any formulas of $\mathcal{L}$ for propositional variables in A and B, respectively,

then

$$A' \equiv B'$$

holds under predicate semantics

Logical Equivalence Example

Example

Consider the following propositional logical equivalence

$$(a \Rightarrow b) \equiv (\neg a \cup b)$$

Substituting

$$\exists x P(x,z) \text{ for } a \quad \text{and} \quad \forall y R(y,z) \text{ for } b$$

we get by the **EquivalencesTheorem** that the following logical equivalence holds

$$(\exists x P(x,z) \Rightarrow \forall y R(y,z)) \equiv (\neg \exists x P(x,z) \cup \forall y R(y,z))$$

Equivalence Substitution

We prove in similar way as in the propositional case the following.

Equivalence Substitution Theorem

Let a formula B_1 be obtained from a formula A_1 by a **substitution** of a formula B for **one** or **more** occurrences of a sub-formula A of A_1, what we denote as

$$B_1 = A_1(A/B)$$

Then the following holds for any formulas A, A_1, B, B_1 of $\mathcal{L}$

$$\text{If} \quad A \equiv B, \quad \text{then} \quad A_1 \equiv B_1$$

Logical Equivalence Theorem

Directly from the Dictum de Omi and theGeneralization tautologies we get the proof of the following theorem useful for building new logical equivalences from the old, known ones

E- Theorem

For any formulas $A(x), B(x)$ of $\mathcal{L}$

$$\text{if } A(x) \equiv B(x), \text{ then } \forall x A(x) \equiv \forall x B(x)$$

$$\text{if } A(x) \equiv B(x), \text{ then } \exists x A(x) \equiv \exists x B(x)$$

Logical Equivalence Example

Example

We know from the previous example that

$$(\exists x P(x,z) \Rightarrow \forall y R(y,z)) \equiv (\neg \exists x P(x,z) \cup \forall y R(y,z))$$

We get, as the direct consequence of the above theorem the following logical equivalence

$$\forall z(\exists x P(x,z) \Rightarrow \forall y R(y,z)) \equiv \forall z(\neg \exists x P(x,z) \cup \forall y R(y,z))$$

$$\exists z(\exists x P(x,z) \Rightarrow \forall y R(y,z)) \equiv \exists z(\neg \exists x P(x,z) \cup \forall y R(y,z))$$

Equational Laws of Quantifiers

We concentrate now only on these laws of quantifiers which have a form of a logical equivalence

They are called the **equational laws** of quantifiers

Directly from the logical equivalence definition and the De Morgan tautologies we get the following

De Morgan Laws

For any formulas $A(x),\ B(x)$ of $\mathcal{L}$

$$\neg\forall x A(x) \equiv \exists x \neg A(x)$$

$$\neg\exists x A(x) \equiv \forall x \neg A(x)$$

We now apply them to show that the **quantifiers** can be defined one by the other i.e. that the following Definability Laws hold

Equational Laws of Quantifiers

Definability Laws

For any formula $A(x)$ of $\mathcal{L}$

$$\forall x A(x) \equiv \neg \exists x \neg A(x)$$

$$\exists x A(x) \equiv \neg \forall x \neg A(x)$$

The first law is often used as a **definition** of the universal quantifier in terms of the existential one (and negation)

The second law is a **definition** of the existential quantifier in terms of the universal one (and negation)

Equational Laws of Quantifiers

Proof of

$$\forall x A(x) \equiv \neg \exists x \neg A(x)$$

Substituting any formula $A(x)$ for a variable a in the propositional equivalence $a \equiv \neg\neg a$
we get by the **Equivalence Theorem** that

$$A(x) \equiv \neg\neg A(x)$$

Applying the **E-Theorem** to the above we obtain

$$\exists x A(x) \equiv \exists x \neg\neg A(x)$$

By the **De Morgan Law**

$$\exists x \neg\neg A(x) \equiv \neg \forall x \neg A(x)$$

By the **Equivalence Substitution Theorem**

$$\exists x A(x) \equiv \neg \forall x \neg A(x)$$

This ends the proof

Equational Laws of Quantifiers

Proof of

$$\forall x A(x) \equiv \neg \exists x \neg A(x)$$

Substituting any formula $A(x)$ for a variable a in the propositional equivalence $a \equiv \neg\neg a$
we get by the **Equivalence Theorem** that

$$A(x) \equiv \neg\neg A(x)$$

Applying the **E-Theorem** to the above we obtain

$$\forall x A(x) \equiv \forall x \neg\neg A(x)$$

By the **De Morgan Law** and **Equivalence Substitution Theorem**

$$\forall x \neg\neg A(x) \equiv \neg \exists x \neg A(x)$$

$$\forall x A(x) \equiv \neg \exists x \neg A(x)$$

This ends the proof

Equational Laws of Quantifiers

Other important equational laws are the following **introduction** and **elimination** laws

Listed equivalences are **not independent**, some of them are the consequences of the others

Introduction and Elimination Laws

If B is a formula such that B **does not** contain any free occurrence of x, then the following logical equivalences hold for any formula $A(x)$ of $\mathcal{L}$

$$\forall x(A(x) \cup B) \equiv (\forall x A(x) \cup B)$$

$$\forall x(A(x) \cap B) \equiv (\forall x A(x) \cap B)$$

$$\exists x(A(x) \cup B) \equiv (\exists x A(x) \cup B)$$

$$\exists x(A(x) \cap B) \equiv (\exists x A(x) \cap B)$$

Equational Laws of Quantifiers

Introduction and Elimination Laws

$$\forall x(A(x) \Rightarrow B) \equiv (\exists x A(x) \Rightarrow B)$$

$$\exists x(A(x) \Rightarrow B) \equiv (\forall x A(x) \Rightarrow B)$$

$$\forall x(B \Rightarrow A(x)) \equiv (B \Rightarrow \forall x A(x))$$

$$\exists x(B \Rightarrow A(x)) \equiv (B \Rightarrow \exists x A(x))$$

As we said before, the equivalences **are not** independent

We show now as an **example** the proof of the third one from the first two

Equational Laws of Quantifiers

We write this proof in a short, symbolic way as follows

$$\exists x(A(x) \cup B) \overset{\text{law}}{\equiv} \neg\forall x \neg(A(x) \cup B)$$
$$\overset{\text{thms}}{\equiv} \neg\forall x(\neg A(x) \cap \neg B)$$
$$\overset{\text{law}}{\equiv} \neg(\forall x \neg A(x) \cap \neg B)$$
$$\overset{\text{law,thm}}{\equiv} (\neg\forall x \neg A(x) \cup \neg\neg B)$$
$$\overset{\text{thm}}{\equiv} (\exists x A(x) \cup B)$$

We leave completion and explanation of all details as it as and **exercise**

Equational Laws of Quantifiers

Distributivity Laws

Let $A(x), B(x)$ be any formulas with a free variable x

Law of distributivity of universal quantifier over conjunction

$$\forall x\,(A(x) \cap B(x)) \equiv (\forall x A(x) \cap \forall x B(x))$$

Law of distributivity of existential quantifier over disjunction

$$\exists x\,(A(x) \cup B(x)) \equiv (\exists x A(x) \cup \exists x B(x))$$

Equational Laws of Quantifiers

Alternations of Quantifiers

Let $A(x, y)$ be any formula with a free variables x, y

$$\forall x \forall y\ (A(x, y)) \equiv \forall y \forall x\ (A(x, y))$$

$$\exists x \exists y\ (A(x, y)) \equiv \exists y \exists x\ (A(x, y))$$

Equational Laws of Quantifiers

Renaming the Variables

Let $A(x)$ be any formula with a free variablex and let y be a variable that **does not occur** in A(x) y, then the following holds

$$\forall x A(x) \equiv \forall y A(y)$$

$$\exists x A(x) \equiv \exists y A(y)$$

Equational Laws of Quantifiers

Restricted De Morgan Laws

For any formulas $A(x), B(x)$ of $\mathcal{L}$

$$\neg\forall_{B(x)}\, A(x) \equiv \exists_{B(x)}\, \neg A(x)$$

$$\neg\exists_{B(x)}\, A(x) \equiv \forall_{B(x)}\neg A(x)$$

Equational Laws of Quantifiers

Here is a poof of first equality

The proof of the second one is similar and is left as an exercise.

$$\neg\forall_{B(x)} A(x) \equiv (\neg\forall x\ (B(x) \Rightarrow A(x)) \equiv$$

$$\neg\forall x\ (\neg B(x) \cup A(x)) \equiv \exists x\ \neg(\neg B(x) \cup A(x)) \equiv$$

$$\exists x\ (\neg\neg B(x) \cap \neg A(x)) \equiv \exists x\ (B(x) \cap \neg A(x)) \equiv \exists_{B(x)} \neg A(x))$$

Restricted Introduction and Elimination Laws

Let B be a formula that **does not** contain any free occurrence of x

then the following logical equivalences hold for any formulas $A(x), B(x), C(x)$ of $\mathcal{L}$

$$\forall_{C(x)}(A(x) \cup B) \equiv (\forall_{C(x)}A(x) \cup B)$$

$$\exists_{C(x)}\ (A(x) \cap B) \equiv (\exists_{C(x)}\ A(x) \cap B)$$

$$\forall_{C(x)}(A(x) \Rightarrow B) \equiv (\exists_{C(x)}A(x) \Rightarrow B)$$

$$\forall_{C(x)}(B \Rightarrow A(x)) \equiv (B \Rightarrow \forall_{C(x)}A(x))$$

The **proofs** are similar to the proof of the restricted De Morgan Laws. The similar generalization of the other Introduction and Elimination Laws for restricted domain quantifiers fails

Equational Laws of Quantifiers

We prove by constructing proper counter-models the following.

$$\exists_{C(x)}(A(x) \cup B) \not\equiv (\exists_{C(x)}A(x) \cup B)$$

$$\forall_{C(x)}(A(x) \cap B) \not\equiv (\forall_{C(x)}A(x) \cap B)$$

$$\exists_{C(x)}(A(x) \Rightarrow B) \not\equiv (\forall_{C(x)}A(x) \Rightarrow B)$$

$$\exists_{C(x)}(B \Rightarrow A(x)) \not\equiv (B \Rightarrow \exists x A(x))$$

Equational Laws of Quantifiers

Nevertheless it is possible to correctly generalize them all as to cover quantifiers with **restricted domain**

We show now how we get the correct generalization of

$$\exists_{C(x)}(A(x) \cup B) \not\equiv (\exists_{C(x)} A(x) \cup B)$$

We leave the other cases an **exercise**

Equational Laws of Quantifiers

Example

The correct restricted quantifiers equality is

$$\exists_{C(x)}(A(x) \cup B) \equiv (\exists_{C(x)}A(x) \cup (\exists x\, C(x) \cap B))$$

We derive it as follows.

$$\exists_{C(x)}(A(x) \cup B) \equiv \exists x(C(x) \cap (A(x) \cup B)) \equiv$$

$$\exists x((C(x) \cap A(x)) \cup (C(x) \cap B)) \equiv (\exists x(C(x) \cap A(x)) \cup \exists x(C(x) \cap B))$$

$$\equiv \exists_{C(x)}A(x) \cup (\exists x\, C(x) \cap B))$$

We leave it as an exercise to specify and write references to transformation or equational laws used at each step of the computation

Chapter 8
Classical Predicate Semantics and Proof Systems

Slides Set 3

PART 4: Proof Systems: Soundness and Completeness

Proof Systems: Soundness and Completeness

We adopt now general definitions from chapter 4 concerning **proof systems** to the case of classical first order (predicate) logic

Chapters 4 and 5 contain a great array of examples, exercises, homework problems explaining in a great detail all notions we introduce here for the predicate case

The **examples** and f **exercises** we provide here are not numerous and restricted to the laws of quantifiers

Proof Systems

Given a predicate language

$$\mathcal{L} = \mathcal{L}_{\{\neg,\cap,\cup,\Rightarrow,\neg\}}(\mathbf{P}, \mathbf{F}, \mathbf{C})$$

Any **proof system**

$$S = (\mathcal{L},\ \mathcal{F},\ LA,\ \mathcal{R})$$

is a **predicate** (first order) proof system

The predicate proof system S is a **Hilbert** proof system if the set $\mathcal{R}$ of its rules contains the Modus Ponens rule

$$(MP)\ \frac{A\ ;\ (A \Rightarrow B)}{B}$$

where $A, B \in \mathcal{F}$

Proof Systems

Semantic Link: Logical Axioms LA

We want the set LA of logical axioms to be a non-empty set of classical predicate tautologies, i.e.

$$LA \subseteq \mathbf{T}_p$$

where

$$\mathbf{T}_p = \{A \text{ of } \mathcal{L}_{\{\neg,\cap,\cup,\Rightarrow,\neg\}}(\mathbf{P},\mathbf{F},\mathbf{C}) : \ \models_p A\}$$

We use symbols

$$\models_p, \ \mathbf{T}_p$$

to stress the fact that we talk about predicate language and classical predicate tautologies

Rules of Inference

Semantic Link 2: Rules of Inference $\mathcal{R}$

We want the the **rules** of inference $r \in \mathcal{R}$ of S to preserve truthfulness. Rules that do so are called **sound**

Definition

Given an inference rule $r \in \mathcal{R}$ of the form

$$(r) \quad \frac{P_1 \ ; \ P_2 \ ; \ .... \ ; \ P_m}{C}$$

where $P_1.P_2, \ldots, P_m, C \in \mathcal{F}$

We say that the rule (r) is **sound** if and only if the following condition holds for **all** structures $\mathbf{M} = [U, I]$ for $\mathcal{L}$

$$\text{If } \mathbf{M} \models \{P_1, P_2, .P_m\} \text{ then } \mathbf{M} \models C$$

Exercise

Prove the soundness of the rule

$$(r) \quad \frac{\forall x A(x)}{\exists x A(x)}$$

Proof

Assume that (r) is **not sound**

It means that **there is** a structure $\mathbf{M} = [U, I]$, such that

$$\mathbf{M} \models \forall x A(x) \quad \text{and} \quad \mathbf{M} \not\models \exists x\, A(x)$$

Let $(\mathbf{M}, s) \models \forall x\, A(x)$ and $(\mathbf{M}, s) \not\models \exists x\, A(x)$

It means that $(\mathbf{M}, s') \models A(x)$ for all s' such that s, s' agree on all variables except on x, and it is **not true** that there is s' such that s, s' agree on all variables except on x , and $(\mathbf{M}, s') \models A(x)$

This is impossible and this **contradiction** proves soundness of (r)

Exercise

Prove that the rule

$$(r) \quad \frac{\exists x A(x)}{\forall x A(x)}$$

is **not sound**

Proof

Observe that to prove that the rule (r) is **not sound** we have to provide an example of an instance of a formula $A(x)$ and construct a counter model

Let $A(x)$ be an atomic formula P(x,c), for any $P \in \mathbf{P}$, $\# P = 2$

We take as a counter model a structure

$$\mathbf{M} = (N, \quad P_I :<, \quad c_I : 3)$$

where N is the set of natural numbers

Rules of Inference

Here is a "shorthand" solution

The atomic formula $(\exists x\ P(x, c))$ becomes in

$$\mathbf{M} = (N,\ \ P_I :<,\ \ \ c_I : 3)$$

a **true** mathematical statement (written with logical symbols):

$$\exists n\ n < 3$$

The formula $(\forall x\ P(x, c))$ becomes a mathematical statement

$$\forall n\ n < 3$$

which is an obviously **false** in the set N of natural numbers

This proves that the the rule (r) is **not sound**

Rules of Inference

Definition of Strongly Sound Rule

An inference rule $r \in \mathcal{R}$ of the form

$$(r) \qquad \frac{P_1 \;\; ; \;\; P_2 \;\; ; \;\; .... \;\; ; \;\; P_m}{C}$$

is **strongly sound** if the following condition holds for all structures $\mathbf{M} = [U, I]$ for $\mathcal{L}$

$$\mathbf{M} \models \{P_1, P_2, .P_m\} \quad \text{if and only if} \quad \mathbf{M} \models C$$

We can, and we do state it informally as

(r) is **strongly sound** if and only if $P_1 \cap P_2 \cap \; \dots \; \cap P_m \equiv C$

Example

The sound rule

$$(r1) \quad \frac{\neg \forall x A(x)}{\exists x \neg A(x)}$$

is **strongly sound** by De Morgan Laws

Example

The sound rule

$$(r2) \quad \frac{\forall x A(x)}{\exists x A(x)}$$

is **not strongly sound** by exercise above

Soundness

Definition of Sound Proof System

Given the **predicate** (first order) proof system

$$S = (\mathcal{L},\ \mathcal{F},\ LA,\ \mathcal{R})$$

We say that S is **sound** if the following conditions hold

(1) $LA \subseteq \mathbf{T}_p$

(2) Each rule of inference $r \in \mathcal{R}$ is **sound**

The proof system S is **strongly sound** if the condition **(2)** is replaced by the following condition **(2')**

(2') Each rule of inference $r \in \mathcal{R}$ is **strongly sound**

Soundness Theorem

When we define (develop) a proof system S our first goal is to make sure that it is a "sound" one

It means that that all we **prove** in it is true. The following theorem establishes this goal

Soundness Theorem for S

Given a predicate proof system S

For any $A \in \mathcal{F}$, the following implication holds.

$$\text{If} \quad \vdash_S A \quad \text{then} \quad \models_p A$$

We write it in a more concise form as

$$\mathbf{P}_S \subseteq \mathbf{T}_p$$

Soundness Theorem

Proof of Soundness Theorem

Observe that if we have already proven that S is **sound** as stated in the definition the proof of the implication

$$\text{If } \vdash_S A \quad \text{then} \quad \models_p A$$

is a straightforward application of the mathematical induction over the length of the formal proof of the formula A

It means that in order to prove the Soundness Theorem for a proof system S it is enough to **verify** the two conditions of the soundness definition, i.e. to verify

(1) $LA \subseteq \mathbf{T}_p$ and

(2) each rule of inference $r \in \mathcal{R}$ is **sound**

CompletessTheorem

Proving **Soundness Theorem** for any proof system S is indispensable and moreover, the proof is quite easy

The next step in developing a **logic** (classical predicate logic in our case now) is to answer the following necessary and difficult question

Given a proof system S about which we know that all it **proves** is true (**tautology**)

Can we **prove** *all we know to be true ?*. It means:

Can S prove all **tautologies***?*

Proving the following **theorem** establishes this goal

CompletenessTheorem

Completeness Theorem for S

Given a predicate proof system S

For any $A \in \mathcal{F}$, the following holds

$$\vdash_S A \quad \text{if and only if} \quad \models_p A$$

We write it in a more concise form as

$$\mathbf{P}_S = \mathbf{T}_p$$

CompletenessTheorem

The **Completeness Theorem** consists of two parts

Part 1: **Soundness Theorem**

$$\mathbf{P}_S \subseteq \mathbf{T}_p$$

Part 2: **Completeness part** of the **Completeness Theorem**

$$\mathbf{T}_p \subseteq \mathbf{P}_S$$

CompletenessTheorem

There are many methods and techniques fo r**proving** the CompletenessTheorem

It applies even for classical proof systems (logics) alone

Non-classical logics often require **new** and usually very sophisticated **methods**

CompletenessTheorem

We presented two very different **proofs** of the **Completeness Theorem** for classical propositional Hilbert style proof system in chapter 5

Then we presented yet another very different **constructive** proofs for automated theorem proving systems for classical propositional logic chapter 6

As a next step we present an old, standard proof of the predicate **Completeness Theorem** for Hilbert style proof system for classical logic in the next chapter 9