Logic and Theoretical Foundation of Computer Science

LaTFoCS

Pamela Fleischmann

fpa@informatik.uni-kiel.de

Winter Semester 2019

Kiel University

Dependable Systems Group

Substitutions in Propositional Logic

Motivation

- we know $\varphi_2 \vee \varphi_1 \equiv \varphi_1 \vee \varphi_2$

Motivation

- we know $\varphi_2 \vee \varphi_1 \equiv \varphi_1 \vee \varphi_2$
- $\rightsquigarrow$ if we talk about semantics, we can use LHS or RHS - whatever suits us best

Motivation

- we know $\varphi_2 \vee \varphi_1 \equiv \varphi_1 \vee \varphi_2$
- $\leadsto$ if we talk about semantics, we can use LHS or RHS - whatever suits us best
- more obvious with $\varphi_1 \rightarrow \varphi_2 \equiv \neg\varphi_1 \vee \varphi$ (see additional material)

Motivation

- we know $\varphi_2 \vee \varphi_1 \equiv \varphi_1 \vee \varphi_2$
- $\rightsquigarrow$ if we talk about semantics, we can use LHS or RHS - whatever suits us best
- more obvious with $\varphi_1 \rightarrow \varphi_2 \equiv \neg\varphi_1 \vee \varphi$ (see additional material)
- formally we are applying a substitution

Substitution

Definition

- A partial function $\sigma : \Phi \to \Phi$ which is the identity on all undefined formulae, is called a substitution.

Substitution

Definition

- A partial function $\sigma : \Phi \to \Phi$ which is the identity on all undefined formulae, is called a substitution.
- The parallel application of σ to a formula $\varphi \in \Phi$ is called substituting φ w.r.t. to σ.

Substitution

Definition

- A partial function $\sigma : \Phi \to \Phi$ which is the identity on all undefined formulae, is called a substitution.
- The parallel application of σ to a formula $\varphi \in \Phi$ is called substituting φ w.r.t. to σ.

other notation: instead of $\sigma(\psi) = \chi$ applied to φ we write $\varphi[\chi \leftarrow \psi]$

Substitution

Definition

- A partial function $\sigma : \Phi \to \Phi$ which is the identity on all undefined formulae, is called a substitution.
- The parallel application of σ to a formula $\varphi \in \Phi$ is called substituting φ w.r.t. to σ.

other notation: instead of $\sigma(\psi) = \chi$ applied to φ we write $\varphi[\chi \leftarrow \psi]$

Example:

$$(p \to q) \land (r \lor s)[\neg p \lor q \leftarrow p \to q] = (\neg p \lor q) \land (r \lor s)$$

Substituting logically equivalent formula

Theorem

For $\varphi, \psi_1, \psi_2 \in \Phi$ *with* $\psi_1 \in \mathrm{Sub}(\varphi)$ *and* $\psi_1 \equiv \psi_2$ *we have* $\varphi \equiv \varphi[\psi_2 \leftarrow \psi_1]$

Substituting logically equivalent formula

Theorem

For $\varphi, \psi_1, \psi_2 \in \Phi$ with $\psi_1 \in \mathrm{Sub}(\varphi)$ and $\psi_1 \equiv \psi_2$ we have $\varphi \equiv \varphi[\psi_2 \leftarrow \psi_1]$

Proof.

- we have to prove $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1])$ for an arbitrary interpretation β

Substituting logically equivalent formula

Theorem

For $\varphi, \psi_1, \psi_2 \in \Phi$ with $\psi_1 \in \mathrm{Sub}(\varphi)$ and $\psi_1 \equiv \psi_2$ we have $\varphi \equiv \varphi[\psi_2 \leftarrow \psi_1]$

Proof.

- we have to prove $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1])$ for an arbitrary interpretation β
- by $\psi_1 \equiv \psi_2$ we know $\hat{\beta}(\psi_2) = \hat{\beta}(\psi_1)$

Substituting logically equivalent formula

Theorem

For $\varphi, \psi_1, \psi_2 \in \Phi$ *with* $\psi_1 \in \mathrm{Sub}(\varphi)$ *and* $\psi_1 \equiv \psi_2$ *we have* $\varphi \equiv \varphi[\psi_2 \leftarrow \psi_1]$

Proof.

- we have to prove $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1])$ for an arbitrary interpretation β
- by $\psi_1 \equiv \psi_2$ we know $\hat{\beta}(\psi_2) = \hat{\beta}(\psi_1)$
- we use the tree representation of φ and induce on the depth of ψ_1 is occuring in φ

Substituting logically equivalent formula

Theorem

For $\varphi, \psi_1, \psi_2 \in \Phi$ with $\psi_1 \in \mathrm{Sub}(\varphi)$ and $\psi_1 \equiv \psi_2$ we have $\varphi \equiv \varphi[\psi_2 \leftarrow \psi_1]$

Proof.

- we have to prove $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1])$ for an arbitrary interpretation β
- by $\psi_1 \equiv \psi_2$ we know $\hat{\beta}(\psi_2) = \hat{\beta}(\psi_1)$
- we use the tree representation of φ and induce on the depth of ψ_1 is occuring in φ
- IB: if ψ_1 occurs at depth 0, ψ_1 has to be φ

Substituting logically equivalent formula

Theorem

For $\varphi, \psi_1, \psi_2 \in \Phi$ with $\psi_1 \in \mathrm{Sub}(\varphi)$ and $\psi_1 \equiv \psi_2$ we have $\varphi \equiv \varphi[\psi_2 \leftarrow \psi_1]$

Proof.

- we have to prove $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1])$ for an arbitrary interpretation β
- by $\psi_1 \equiv \psi_2$ we know $\hat{\beta}(\psi_2) = \hat{\beta}(\psi_1)$
- we use the tree representation of φ and induce on the depth of ψ_1 is occuring in φ
- IB: if ψ_1 occurs at depth 0, ψ_1 has to be φ
  - thus $\hat{\beta}(\varphi)[\psi_2 \leftarrow \psi_1] = \hat{\beta}(\psi_1)[\psi_2 \leftarrow \psi_1] = \hat{\beta}(\psi_2) = \hat{\beta}(\psi_1) = \hat{\beta}(\varphi)$.

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$
- IS: since $d > 0$, $\varphi = \neg\chi$ or $\varphi = \chi_1 \circ \chi_2$ where $\circ$ is one of the binary operators

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$
- IS: since $d > 0$, $\varphi = \neg\chi$ or $\varphi = \chi_1 \circ \chi_2$ where $\circ$ is one of the binary operators
- thus ψ_1 is a subformula of χ, χ_1 or χ_2 and it occurs therein at depth $d - 1$.

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$
- IS: since $d > 0$, $\varphi = \neg\chi$ or $\varphi = \chi_1 \circ \chi_2$ where $\circ$ is one of the binary operators
- thus ψ_1 is a subformula of χ, χ_1 or χ_2 and it occurs therein at depth $d - 1$.
- By IH: $\hat{\beta}(\chi_1) = \hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]$ and analogously for χ_2 and χ

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$
- IS: since $d > 0$, $\varphi = \neg\chi$ or $\varphi = \chi_1 \circ \chi_2$ where $\circ$ is one of the binary operators
- thus ψ_1 is a subformula of χ, χ_1 or χ_2 and it occurs therein at depth $d - 1$.
- By IH: $\hat{\beta}(\chi_1) = \hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]$ and analogously for χ_2 and χ
- this implies $\hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]) = \hat{\beta}((\neg\chi_1)[\psi_2 \leftarrow \psi_1])$ which is `true` iff $\hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]) =$ `false`.

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$
- IS: since $d > 0$, $\varphi = \neg\chi$ or $\varphi = \chi_1 \circ \chi_2$ where $\circ$ is one of the binary operators
- thus ψ_1 is a subformula of χ, χ_1 or χ_2 and it occurs therein at depth $d - 1$.
- By IH: $\hat{\beta}(\chi_1) = \hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]$ and analogously for χ_2 and χ
- this implies $\hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]) = \hat{\beta}((\neg\chi_1)[\psi_2 \leftarrow \psi_1])$ which is $\texttt{true}$ iff $\hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]) = \texttt{false}$.
- the latter holds iff $\hat{\beta}(\chi_1) = \texttt{false}$ and this is equivalent to $\hat{\beta}(\varphi) = \texttt{true}$

Proof (Cont.)

- IH: Assume that $\hat{\beta}(\varphi) = \hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]$ if ψ_1 occurs at depth $d > 0$ for an arbitrary but fixed $d \in \mathbb{N}$
- IS: since $d > 0$, $\varphi = \neg\chi$ or $\varphi = \chi_1 \circ \chi_2$ where $\circ$ is one of the binary operators
- thus ψ_1 is a subformula of χ, χ_1 or χ_2 and it occurs therein at depth $d - 1$.
- By IH: $\hat{\beta}(\chi_1) = \hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]$ and analogously for χ_2 and χ
- this implies $\hat{\beta}(\varphi[\psi_2 \leftarrow \psi_1]) = \hat{\beta}((\neg\chi_1)[\psi_2 \leftarrow \psi_1])$ which is $\mathtt{true}$ iff $\hat{\beta}(\chi_1[\psi_2 \leftarrow \psi_1]) = \mathtt{false}$.
- the latter holds iff $\hat{\beta}(\chi_1) = \mathtt{false}$ and this is equivalent to $\hat{\beta}(\varphi) = \mathtt{true}$
- the other cases are analogous $\square$

Rules for Logical Equivalence (proofs left to the reader)

- `true` is the neutral element of $\land$ and $\leftrightarrow$, `false` is the neutral element of $\lor$ and $\dot\lor$

Rules for Logical Equivalence (proofs left to the reader)

- $\texttt{true}$ is the neutral element of $\wedge$ and $\leftrightarrow$, $\texttt{false}$ is the neutral element of $\vee$ and $\dot{\vee}$
- $\varphi \vee \texttt{true} \equiv \texttt{true}$, $\varphi \wedge \texttt{false} \equiv \texttt{false}$

Rules for Logical Equivalence (proofs left to the reader)

- `true` is the neutral element of $\wedge$ and $\leftrightarrow$, `false` is the neutral element of $\vee$ and $\dot{\vee}$
- $\varphi \vee \texttt{true} \equiv \texttt{true}$, $\varphi \wedge \texttt{false} \equiv \texttt{false}$
- $\varphi \rightarrow \texttt{true} \equiv \texttt{true}$, $\varphi \rightarrow \texttt{false} \equiv \neg\varphi$

Rules for Logical Equivalence (proofs left to the reader)

- $\mathtt{true}$ is the neutral element of $\wedge$ and $\leftrightarrow$, $\mathtt{false}$ is the neutral element of $\vee$ and $\dot{\vee}$
- $\varphi \vee \mathtt{true} \equiv \mathtt{true}$, $\varphi \wedge \mathtt{false} \equiv \mathtt{false}$
- $\varphi \rightarrow \mathtt{true} \equiv \mathtt{true}$, $\varphi \rightarrow \mathtt{false} \equiv \neg\varphi$
- $\mathtt{true} \rightarrow \varphi \equiv \varphi$, $\mathtt{false} \rightarrow \varphi \equiv \mathtt{true}$ (ex falso quodlibet)

Rules for Logical Equivalence (proofs left to the reader)

- $\texttt{true}$ is the neutral element of $\wedge$ and $\leftrightarrow$, $\texttt{false}$ is the neutral element of $\vee$ and $\dot{\vee}$
- $\varphi \vee \texttt{true} \equiv \texttt{true}$, $\varphi \wedge \texttt{false} \equiv \texttt{false}$
- $\varphi \rightarrow \texttt{true} \equiv \texttt{true}$, $\varphi \rightarrow \texttt{false} \equiv \neg\varphi$
- $\texttt{true} \rightarrow \varphi \equiv \varphi$, $\texttt{false} \rightarrow \varphi \equiv \texttt{true}$ (ex falso quodlibet)
- $\varphi \leftrightarrow \texttt{false} \equiv \neg\varphi$

Rules for Logical Equivalence (proofs left to the reader)

- $\mathtt{true}$ is the neutral element of $\wedge$ and $\leftrightarrow$, $\mathtt{false}$ is the neutral element of $\vee$ and $\dot{\vee}$
- $\varphi \vee \mathtt{true} \equiv \mathtt{true}$, $\varphi \wedge \mathtt{false} \equiv \mathtt{false}$
- $\varphi \rightarrow \mathtt{true} \equiv \mathtt{true}$, $\varphi \rightarrow \mathtt{false} \equiv \neg\varphi$
- $\mathtt{true} \rightarrow \varphi \equiv \varphi$, $\mathtt{false} \rightarrow \varphi \equiv \mathtt{true}$ (ex falso quodlibet)
- $\varphi \leftrightarrow \mathtt{false} \equiv \neg\varphi$
- $\varphi \dot{\vee} \mathtt{true} \equiv \neg\varphi$

Rules for Logical Equivalence (proofs left to the reader)

- $\vee$ and $\wedge$ are idempotent

Rules for Logical Equivalence (proofs left to the reader)

- $\lor$ and $\land$ are idempotent
- $\varphi \equiv \neg\neg\varphi$

Rules for Logical Equivalence (proofs left to the reader)

- $\vee$ and $\wedge$ are idempotent
- $\varphi \equiv \neg\neg\varphi$
- $\varphi \vee \neg\varphi$, $\varphi \rightarrow \varphi$, $\varphi \leftrightarrow \varphi$ are tautologies

Rules for Logical Equivalence (proofs left to the reader)

- $\vee$ and $\wedge$ are idempotent
- $\varphi \equiv \neg\neg\varphi$
- $\varphi \vee \neg\varphi$, $\varphi \rightarrow \varphi$, $\varphi \leftrightarrow \varphi$ are tautologies
- $\varphi \wedge \neg\varphi$, $\varphi \dot{\vee} \varphi$ are contradictions

Rules for Logical Equivalence (proofs left to the reader)

- all binary operators but the implication are commutative

Rules for Logical Equivalence (proofs left to the reader)

- all binary operators but the implication are commutative
- disjunction, conjunction, equivalence, and xor are associative

Rules for Logical Equivalence (proofs left to the reader)

- all binary operators but the implication are commutative
- disjunction, conjunction, equivalence, and xor are associative
- disjunction and conjunction are distributive

Rules for Logical Equivalence (proofs left to the reader)

- all binary operators but the implication are commutative
- disjunction, conjunction, equivalence, and xor are associative
- disjunction and conjunction are distributive
- contraposition: $\varphi \rightarrow \psi \equiv \neg\psi \rightarrow \neg\varphi$

Much ado about nothing . . .

- what do we have to do, if we prove claims for propositional logic formula?
  - we take one arbitrary and use structural induction
- can we always generalise the binary operators to $\circ$ in a proof?
  - no! they have different properties
- it would be great if some of them are expressible by the others

Much ado about nothing . . .

- what do we have to do, if we prove claims for propositional logic formula?
  - we take one arbitrary and use structural induction
  - thus we have a case for each operator at depth 0
- can we always generalise the binary operators to $\circ$ in a proof?
  - no! they have different properties
- it would be great if some of them are expressible by the others

Expressive Power of Fragments of Propositional Logic

$S = \{\wedge, \vee, \neg, \dot{\vee}, \rightarrow, \leftrightarrow, \uparrow, \downarrow\}$

Definition

- The expressive power of Φ is the set of all formula φ such there does not exist a $\psi \in \Phi$ with $\psi \equiv \phi$.

Expressive Power of Fragments of Propositional Logic

$S = \{\wedge, \vee, \neg, \dot{\vee}, \rightarrow, \leftrightarrow, \uparrow, \downarrow\}$

Definition

- The expressive power of Φ is the set of all formula φ such there does not exist a $\psi \in \Phi$ with $\psi \equiv \phi$.
- For $T \subseteq S$, let Φ_T be the set of all formula only containing operators from T. The expressive power of Φ_T is defined as above.

$S = \{\wedge, \vee, \neg, \dot{\vee}, \rightarrow, \leftrightarrow, \uparrow, \downarrow\}$

Definition

- The expressive power of Φ is the set of all formula φ such there does not exist a $\psi \in \Phi$ with $\psi \equiv \phi$.
- For $T \subseteq S$, let Φ_T be the set of all formula only containing operators from T. The expressive power of Φ_T is defined as above.
- Two sets Φ_T and Φ_R for $T, R \subseteq S$ have the same expressive power if they define the same formula up to logical equivalence.

Expressive Power (Example)

We saw $\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi$!

Expressive Power (Example)

We saw $\varphi \to \psi \equiv \neg\varphi \vee \psi$!

- set $T := \{\wedge, \vee, \neg, \dot{\vee}, \leftrightarrow, \uparrow, \downarrow\}$

Expressive Power (Example)

We saw $\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi$!

- set $T := \{\wedge, \vee, \neg, \dot{\vee}, \leftrightarrow, \uparrow, \downarrow\}$
- Φ_T and Φ have the same expressive power

Expressive Power (Example)

We saw $\varphi \rightarrow \psi \equiv \neg\varphi \lor \psi$!

- set $T := \{\land, \lor, \neg, \dot{\lor}, \leftrightarrow, \uparrow, \downarrow\}$
- Φ_T and Φ have the same expressive power
- sketch of proof: $\Phi_T \subseteq \Phi$

Expressive Power (Example)

We saw $\varphi \to \psi \equiv \neg\varphi \vee \psi$!

- set $T := \{\wedge, \vee, \neg, \dot{\vee}, \leftrightarrow, \uparrow, \downarrow\}$
- Φ_T and Φ have the same expressive power
- sketch of proof: $\Phi_T \subseteq \Phi$
- for each $\varphi \in \Phi$ that contains $\to$ we apply the substitution $\sigma(p \to q) = \neg p \vee q$ for each $p \to q$ in φ

Expressive Power (Example)

We saw $\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi$!

- set $T := \{\wedge, \vee, \neg, \dot{\vee}, \leftrightarrow, \uparrow, \downarrow\}$
- Φ_T and Φ have the same expressive power
- sketch of proof: $\Phi_T \subseteq \Phi$
- for each $\varphi \in \Phi$ that contains $\rightarrow$ we apply the substitution $\sigma(p \rightarrow q) = \neg p \vee q$ for each $p \rightarrow q$ in φ
- after the application $\varphi[\sigma]$ is in Φ_T

Logical Equivalence for Removing Operators

$$
\begin{aligned}
\varphi \leftrightarrow \psi &\equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi), & \varphi \rightarrow \psi &\equiv \neg\varphi \vee \psi \\
\varphi \vee \psi &\equiv \neg(\neg\varphi \wedge \neg\psi), & \varphi \wedge \psi &\equiv \neg(\neg\varphi \vee \neg\psi), \\
\varphi \dot{\vee} \psi &\equiv \neg(\varphi \rightarrow \psi) \vee \neg(\psi \rightarrow \varphi), & & \\
\varphi \uparrow \psi &\equiv \neg(\varphi \wedge \psi), & \varphi \downarrow \psi &\equiv \neg(\varphi \vee \psi).
\end{aligned}
$$

Choosing a Set of Operators

Definition

Let S be a set of operators. The binary operator $\circ$ is defined from S if for all $\varphi_1, \varphi_2 \in \Phi_S$ there exists $\psi \in \Phi_S$ with $\varphi_1 \circ \varphi_2 \equiv \psi$. The unary operator $\star$ is defined from S if for all $\varphi \in \Phi_S$ there exists $\psi \in \Phi_S$ with $\star\varphi \equiv \psi$.

Choosing a Set of Operators

Definition

Let S be a set of operators. The binary operator $\circ$ is defined from S if for all $\varphi_1, \varphi_2 \in \Phi_S$ there exists $\psi \in \Phi_S$ with $\varphi_1 \circ \varphi_2 \equiv \psi$. The unary operator $\star$ is defined from S if for all $\varphi \in \Phi_S$ there exists $\psi \in \Phi_S$ with $\star\varphi \equiv \psi$.

Roughly spoken: an operator is defined from a set, if we can express it by operators from S.

Set of Operators used mostly in Logic

Theorem

Set $O = \{\neg, \circ\}$ for $\circ \in \{\wedge, \vee, \rightarrow\}$. Then all other operators introduced here can be defined from O.

Set of Operators used mostly in Logic

Theorem

Set $O = \{\neg, \circ\}$ for $\circ \in \{\wedge, \vee, \rightarrow\}$. Then all other operators introduced here can be defined from O.

Proof.

- follows directly by the equivalence rules $\square$

Set of Operators used mostly in Logic

Theorem

Set $O = \{\neg, \circ\}$ for $\circ \in \{\wedge, \vee, \rightarrow\}$. Then all other operators introduced here can be defined from O.

Theorem

The set O is minimal.

The proof is left to the reader.

Why do we have nand or nor gates?

Theorem

If $\circ$ is a binary operator that can define negation and all other binary operators introduced here, by itself, then $\circ$ is either $\uparrow$ or $\downarrow$.

but

- we don't think in nand or nor: *I don't do not buying milk and bread and not buying milk and bread* $\sim (M \uparrow B) \uparrow (M \uparrow B)$

Why do we have nand or nor gates?

Theorem

If $\circ$ is a binary operator that can define negation and all other binary operators introduced here, by itself, then $\circ$ is either $\uparrow$ or $\downarrow$.

assuming that the theorem is true (the proof is omitted here),

- nand and nor are very powerful
- for a computer only one device has to be constructed
- it is cheap

but

- we don't think in nand or nor: *I don't do not buying milk and bread and not buying milk and bread* $\sim (M \uparrow B) \uparrow (M \uparrow B)$