Peter Koepke

GÖDEL'S COMPLETENESS THEOREM WITH NATURAL LANGUAGE FORMULAS*

ABSTRACT

We give a self-contained proof of the GÖDEL completeness theorem based on "natural language". Utilizing a naive understanding of language, the semantics of natural language formulas is intuitively clear, which makes the correctness of our natural deduction style proof calculus immediate. The converse direction, including a HENKIN-style model construction, is more involved, but hopefully "natural" as well.

1 INTRODUCTION

One of KURT GÖDEL's great achievements is the *completeness theorem* for first-order logic, which he proved in his 1929 Vienna doctoral dissertation *Über die Vollständigkeit des Logikkalküls* (published as [2]). The completeness theorem can be seen as the *fundamental theorem* of mathematical logic, showing the universality of the mathematical proof method. Mathematical proofs consist (in principle) of a sequence of elementary logical steps. The completeness theorem states that *every* universally valid statement is formally provable.

The completeness theorem also has a bearing on the famous GÖDEL *incompleteness theorems* [3]. The incompleteness of theories like PEANO arithmetic can only be appreciated in contrast to the completeness of the underlying logic.

At the University of Bonn, the GÖDEL centenary was commemorated by two special lectures of the *Dies Academicus* in May 2006. Prof. Dr. RAINER STUHLMANN-LAEISZ gave a talk "Unbeweisbare Wahrheiten - zum 100. Geburstag von Kurt Gödel" on the GÖDEL incompleteness theorems. The current paper is an elaboration of the second lecture which presented the GÖDEL completeness theorem to a general academic audience.

**Dedicated to Prof. Dr.* RAINER STUHLMANN-LAEISZ *on the occasion of his 65. birthday*

2 Example: the irrationality of $\sqrt{2}$

In ordinary, "semi-formal" mathematical proofs technical formulas are connected by natural language phrases which give information on logical dependencies, on proof methods, and on intuitions intended to enable the reader to supply the missing details. By filling in gaps and normalizing the language one arrives at proofs where the individual steps are of a very elementary nature.

Let us consider a standard proof of the irrationality of $\sqrt{2}$. The proof presupposes some elementary facts from arithmetic which we do not state in detail. We number the lines of the proof for later reference.

Theorem 1 *$\sqrt{2}$ is irrational.*

Proof

1 Assume $\sqrt{2}$ is rational.
2 Assume that $\sqrt{2} = \frac{a}{b}$, and that
3 a is even implies that b is odd.
4 $2 = \frac{a}{b} \cdot \frac{a}{b} = \frac{a \cdot a}{b \cdot b}$.
5 $2 \cdot b \cdot b = a \cdot a$.
6 *Case 1.* Assume a is odd.
7 $2 \cdot b \cdot b$ is even.
8 $a \cdot a$ is even.
9 $a \cdot a$ is odd.
10 Contradiction.
11 Thus
12 a is odd implies a contradiction.
13 *Case 2.* Assume a is not odd.
14 a is even.
15 b is odd.
16 $b \cdot b = \frac{a}{2} \cdot a$. $b \cdot b$ is odd. $\frac{a}{2} \cdot a$ is even.
17 Contradiction.
18 Thus
19 a not odd implies a contradiction.
20 Contradiction.
21 Thus
22 $\sqrt{2}$ is rational implies a contradiction.
23 $\sqrt{2}$ is not rational. □

Most arguments in this proof are of a purely formal nature, i.e., they only depend upon the syntactic structure of assumptions and conclusions regardless of the semantic meaning of the statements:

– Line 7: "$2 \cdot b \cdot b$ is even" follows from the number theoretic background assumptions. The next line 8: "$a \cdot a$ is even", can be produced from lines 7 and 5 simply by replacing $2 \cdot b \cdot b$ by $a \cdot a$. This corresponds to the transformation

  $$\frac{2 \cdot b \cdot b \text{ is even} \quad 2 \cdot b \cdot b = a \cdot a}{a \cdot a \text{ is even}}.$$

  Generally this *substitution rule* can be expressed as

  $$\frac{t \text{ has the property } A \quad t = t'}{t' \text{ has the property } A}.$$

– Line 10: "contradiction" follows from lines 8 and 9, since one is the negation of the other when we take "odd" to be an abbreviation of "not even". The *contradiction rule* can be expressed by

  $$\frac{A \quad \text{not } A}{\text{contradiction}}.$$

– Line 15: "*b* is odd" is a logical consequence of line 14: "*a* is even", and line 3: "*a* is even implies that *b* is odd". This uses the rule *modus ponens*

  $$\frac{A \quad A \text{ implies } B}{B}.$$

In the next section we shall exhibit a proof calculus consisting of similar rules.

Usually in the course of a proof, assumptions are introduced to carry out "subproofs" which depend on those extra assumptions. Consider the case distinction of the example:

– Line 6: "*Case 1.* Assume *a* is odd" introduces an assumption leading to line 10: "contradiction". The subargument is then closed by the keyword "thus" in line 11. The subsequent line 12, draws the conclusion "*a* is odd implies a contradiction" from the subargument.

The "assume-thus" construct structures the proof into (nested) (sub-)arguments. As in the example one may visualize this structure by *indentations*: "Assume *A*" starts an indentation and adds *A* to the assumptions usable in the course of the subargument. “Thus” finishes a subargument and also the present indentation. Directly after the subargument one may introduce an implication “A implies B” where B is the last formula of the subargument before “Thus”. After that the subargument becomes invisible for the further course of the argument.

3 A NATURAL MATHEMATICAL LANGUAGE

We give a formalization of the mathematical language used in the $\sqrt{2}$-example. We shall work with natural language constructs like “for all x” instead of introducing formal quantifiers “$\forall x$”. In this way the meaning or the *semantics* of formulas becomes self-explanatory. For the purposes of this paper we keep the language small, allowing for small definitions, a small proof calculus, and a small number of cases in the completeness proof. On the other hand one may imagine an extended mathematical language built on similar principles, which models many common phrases, and an associated rich proof calculus, which contains common figures of argument [6]. The NaProChe project [7] is developing that approach.

Our language is based on a sufficiently large reservoir of basic symbols. Since functions can be modeled by relations, we can omit functions from the language. Also equality ($=$) can be seen as another binary relation. So we assume that for every natural number n there is a countable supply of n-ary relations available which we denote by $R(x_1,\ldots,x_n)$, $S(x_1,\ldots,x_n)$, ...

For simplicity we also restrict the number of logical connectives. Since “A and B” is equivalent to “not(not A or not B)” one obtains “and” from “not” and “or”; since “A or B” is equivalent to “not A implies B” one obtains “or” from “not” and “implies”. So we only allow “not” and “implies” as propositional connectives. Since “there is x such that A” is equivalent to “not for all x holds not A” we can restrict to the universal quantifier “for all”.

Definition 1 *The collection of (natural language) formulas is defined by:*

– *every* relational formula $R(x_1,\ldots,x_n)$ *is a formula; for specific relations like “odd” or “$=$” one may also write as usual “x is odd” or “$x < y$” instead of $R(x)$ or $R(x,y)$;*

– *“contradiction” is a formula;*

– *if A is a formula then “not A” is a formula;*

- *if A and B are formulas then "(A implies B)" is a formula;*
- *if A is a formula then "for all x (holds) A" is a formula.*

Brackets in formulas may be omitted according to the usual conventions.

Any natural language formula is making statements about a finite set of *free*, or *global* variables. This set is defined recursively.

Definition 2 *For a formula A define the set* $\text{free}(A)$ *of* free variables *of A by recursion:*

- $\text{free}(R(x_1,\ldots,x_n)) = \{x_1,\ldots,x_n\}$;
- $\text{free}(\text{contradiction}) = \emptyset$;
- $\text{free}(\text{not}A) = \text{free}(A)$;
- $\text{free}(A\,\text{implies}\,B) = \text{free}(A) \cup \text{free}(B)$;
- $\text{free}(\text{for all}\,x\,\text{holds}\,A) = \text{free}(A) \setminus \{x\}$.

Thus the "local" variable x of the quantification "for all x holds A" is not free after the quantification. If Φ *is a set of formulas then* $\text{free}(\Phi)$ *is the set of all variables which occur free in some formula in* Φ.

E.g., the transitivity axiom

$$\text{for all } x \text{ for all } y \text{ for all z } (x < y \text{ implies } (y < z \text{ implies } x < z))$$

has no free variable since all occurring variables are quantified.

Mathematical formulas can be *interpreted* in suitable structures. Given a structure $(A, <^A)$ with a binary relation $<^A$ on A one can canonically check whether the above transitivity axiom *holds* in $(A, <^A)$.

Definition 3 *A formula is called* universally valid *if it holds in all suitable structures.*

The GÖDEL completeness theorem states that every universally valid formula is formally provable.

4 PROOFS

We work with a GENTZEN-style natural deduction calculus [1], using a small number of basic proof rules.

Definition 4 *The* basic proof rules *are given by the rules of*

a)	*contradiction*	$\dfrac{A \qquad \text{not } A}{\text{contradiction}}$
b)	*proof by contradiction*	$\dfrac{\text{not } A \text{ implies contradiction}}{A}$
c)	*modus ponens*	$\dfrac{A \text{ implies } B \qquad A}{B}$
d)	*instantiation*	$\dfrac{\text{for all } x \text{ holds } A(x)}{A(y)}$
e)	*generalization*	$\dfrac{A(y)}{\text{for all } x \text{ holds } A(x)}$

Note that the first four rules are *correct* in the following sense: if the assumptions of the rule hold in some structure then the conclusion also holds in the structure. The situation is more complex for the generalization rule which can only be applied in certain proof situations.

Definition 5 *A* (mathematical) text *is a sequence $T = S_1 \dots S_l$ of* statements *where each statement is of the form $S_k =$"Assume A_k.", $S_k =$"A_k.", or $S_k =$"Thus" for some formula A_k.*

The proof of the irrationality of $\sqrt{2}$ is basically a mathematical text in the sense of this definition. A text is a *proof* if every line within the text is formally justified, e.g., that it can be generated by a proof rule from previous lines which are "visible" to the present line. Visibility can be calculated via *indentation depths*: a previous line is visible if it is not "blocked" by some "Thus" which has the same indentation level as that previous line. This is formalized by the following definitions.

Definition 6 *Let $T = S_1 \dots S_l$ be a mathematical text. Then define:*

a) For $k \leqslant l$ let

$$\begin{aligned} \text{ind}_T(k) \quad = \quad & |\{j \leqslant k \mid S_j \text{ starts with "Assume"}\}| - \\ & |\{j < k \mid S_j \text{ starts with "Thus"}\}| \end{aligned}$$

be the (indentation) depth *of* S_k *in* T. *It is given by the difference between the numbers of previous assumptions ("Assume") and the previous conclusions ("Thus").*

b) *The text* T *is* properly indented *if* $\text{ind}_T(k) \geqslant 0$ *for all* $k \leqslant l$, *i.e., we cannot have more conclusions than assumptions.*

c) *For* $i < k \leqslant l$ *the line number* i *is* visible *from line number* k *if there is no* j, $i \leqslant j < k$ *such that* $S_j =$*"Thus" and* $\text{ind}_T(i) = \text{ind}_T(j)$. *In case* i *is visible from* k *we also say that the formula* A_i *and the free variables of* A_i *are visible from* k.

Definition 7 *Let* $T = S_1 \ldots S_l$ *be a mathematical text. Let* Φ *be a set of formulas.*

a) T *is a* (formal) proof *from* Φ *if* T *is properly indented, and for all* $k \leqslant l$ *one of the following holds:*

   i. $S_k =$*"Assume* A_k*.", or* $S_k =$*"Thus"; this means that we can introduce an assumption or try to conclude a subargument at any place in a proof;*

   ii. $S_k =$*"*A_k*." where* $A_k \in \Phi$ *or* $A_k = A_i$, *i<k where the line* S_i *is* visible *by* S_k; *this means that the "axioms" contained in* Φ *or visible statements established previously can be used freely;*

   iii. $S_k =$*"*A_k*." where* A_k *can be produced by one of the basic proof rules from formulas which are elements of* Φ *or which are visible from* k; *moreover, if* A_k *is of the form* $A_k =$*"for all* x *holds* $A(x)$*" and is produced by the rule of generalization from the formula* $A(y)$, *we also require that the variable* $y \notin \text{free}(\Phi)$ *and that* y *is not visible from* k *as a free variable; so the generalization from* $A(y)$ *to "for all* x *holds* $A(x)$*" is possible if* y *was a "general" variable without further specifications in* Φ *or previous relevant formulas;*

   iv. $S_k =$*"*A_i *implies* A_{k-2}*" where* $S_{k-1} =$*"Thus" and* $i \leqslant k-2$ *is the minimal line number which is visible from* $k-1$; *we say that* S_k *is produced by the* rule of implication; *the result of a subargument from the assumption* A_i *to the conclusion* A_{k-2} *is the implication "*A_i *implies* A_{k-2}*".*

b) T *is a* (formal) proof of A *from* Φ *if* $A = A_l$ *and* $\text{ind}_T(l) = 0$; *the latter means that all subarguments have been concluded.*

c) *A is* (formally) provable *from* Φ *if there exists a proof of A from* Φ.

d) *A is* (formally) provable *if it is provable from the empty set* $\emptyset$*, i.e., without further hypothesis.*

We demonstrate this type of (formal) proof by proving some *derived rules* which may also be used conveniently in further proofs.

Proposition 1 *Let* A, B *be formulas. Then A is provable from the formulas "B implies A", "(not B) implies A". This justifies the use of the* derived rule *of* case distinction*:*

$$\frac{B \text{ implies } A \quad \text{not } B \text{ implies } A}{A}.$$

Proof The following is a proof of A from $F_1 =$ "B implies A" and $F_2 =$ "not B implies A". We also state the rules which are applied and the local depths and hypotheses.

k	Statement	Rule ... with hypothesis ...	$\text{ind}_T(k)$	visible lines
1	Assume not A.	-	1	-
2	Assume not B.	-	2	1
3	A.	*modus ponens* w. 2, F_2	2	1,2
4	Contradiction.	*contradiction* w. 1, 3	2	1,2,3
5	Thus	-	2	1,2,3,4
6	not B implies a contradiction.	*implication*	1	1
7	B.	*proof by contradiction* w. 6	1	1,6
8	A.	*modus ponens* w. 7, F_1	1	1,6,7
9	Contradiction.	*contradiction* w. 1, 8	1	1,6,7,8
10	Thus	-	1	1,6,7,8,9
11	not A implies a contradiction.	*implication*	0	-
12	A.	*proof by contradiction* w. 11	0	11

□

Proposition 2 *Let A be a formula. Then A is provable from the formula "contradiction". This justifies the use of the* derived rule *of* ex falsum libenter*:*

$$\frac{\text{contradiction}}{A}.$$

Proof The following is a formal proof of A from $F =$ "contradiction".

k	Statement	Rule ... with hypothesis ...	$\text{ind}_T(k)$	visible lines
1	Assume not A.	-	1	-
2	Contradiction.	Copying F	1	1
3	Thus	-	1	2
4	not A implies a contradiction.	*implication*	0	-
5	A.	*proof by contradiction* with 4	0	4

□

A formal proof as defined in Definition 7, though formulated in a "poor" vocabulary and grammar, can be read as a proof in the ordinary mathematical sense. Since mathematical proofs prove universally valid statements, we obtain the *correctness theorem*:

Theorem 2 *If a formula A is provable then it is universally valid.*

5 THE COMPLETENESS THEOREM

GÖDEL's *completeness theorem* is the converse of the correctness theorem. Our proof uses the approach of L. HENKIN [5] and G. HASENJAEGER [4].[1] Given a formula which is not provable build a HENKIN set of formulas (denoted by $\mathscr{H}$ in the subsequent proof) which describes a structure in which A fails. Then build such a structure $\mathscr{S}$ out of the terms of the language.

Theorem 3 *If a formula A is universally valid it is provable.*

Proof Assume that A is not provable. It suffices to show that A is not universally valid by constructing a structure $\mathscr{S}$ in which A does not hold.

We shall recursively define a sequence $A_1, A_2, A_3, \ldots$ of formulas which describe the structure $\mathscr{S}$. Along the recursion we maintain that A is not provable from $A_1, \ldots, A_n$. To extend the sequence, we postulate two extension properties: by (1), every formula can be decided positively or negatively in the construction; by (2), we can add a counterexample to every universal formula which is not valid.

[1]Prof. Dr. Gisbert Hasenjaeger (1919 – 2006) prededed Prof. Dr. Rainer Stuhlmann-Laeisz from 1962 until 1984 on the chair for logic and foundations at the University of Bonn.

(1) Assume that A is not provable from $A_1,\dots,A_n$ and let B be a formula. Then A is not provable from $A_1,\dots,A_n,B$, *or* A is not provable from $A_1,\dots,A_n$, "not B".

Proof. Assume not. Assume that the mathematical text Proof1, A is a proof of A from $A_1,\dots,A_n,B$ *and* that Proof2, A is a proof of A from $A_1,\dots,A_n$, "not B". Then the following combined text is a proof of A from $A_1,\dots,A_n$:

k	Statement	Rule ... with hypothesis ...
1	Assume B.	-
2	Proof1	given
3	A	given
4	Thus	-
5	B implies A	*implication*
6	Assume not B	-
7	Proof2	given
8	A	given
9	Thus	-
10	not B implies A	*implication*
11	A	*case distinction* with 5, 10

This contradicts the initial assumption. *qed*(1)

(2) Assume that A is not provable from

$$A_1,\dots,A_n,\text{“not for all } x \text{ holds } B(x)\text{”}$$

and that y is a variable which does not occur in $A_1,\dots,A_n$, or in the formula "not for all x holds $B(x)$". Then A is not provable from

$$A_1,\dots,A_n,\text{“not for all } x \text{ holds } B(x)\text{”},\text{“not } B(y)\text{”}.$$

Proof. Assume not and assume that the text Proof1, A is a proof of A from

$$A_1,\dots,A_n,\text{“not for all } x \text{ holds } B(x)\text{”},\text{“not } B(y)\text{”}.$$

Then the following combined text is a proof of A from $A_1,\dots,A_n$, and "not for all x holds $B(x)$":

k	Statement	Rule ... with hypothesis ...
1	Assume $B(y)$.	-
2	For all x holds $B(x)$.	*generalization* with 1
3	Contradiction.	*contradiction* with 2 and "not for all x holds $B(x)$"
4	A	*ex falsum libenter* with 3
5	Thus	-
6	$B(y)$ implies A.	*implication*
7	Assume not $B(y)$.	-
8	Proof1	given
9	A.	given
10	Thus	-
11	not $B(y)$ implies A.	*implication*
12	A.	*case distinction* with 6, 11

This contradicts the initial assumption. *qed*(2)

The collection of formulas is countable since every formula is basically a finite sequence of symbols taken from a countable or even finite alphabet. Let $F_1, F_2, \ldots$ be an enumeration of all formulas.

Define a sequence $A_1, A_2, \ldots$ of formulas by recursion. At odd stages $1, 3, \ldots$, we ensure that every formula is decided by the sequence; at even stages $2, 4, 6, \ldots$, we care about quantifiers. So let $2m-1$ be an odd number, where $m \geqslant 1$, and assume that $A_1, \ldots, A_{2m-2}$ are defined. We shall define A_{2m-1} and A_{2m}.

If A is not provable from $\{A_1, \ldots, A_{2m-2}, F_m\}$, set $A_{2m-1} = F_m$; otherwise set $A_{2m-1} =$ "not F_m". Thereafter, if A_{2m-1} is of the form "not for all x holds $B(x)$", choose a variable y which does not occur in $\{A_0, \ldots, A_{2m-1}\}$ and set $A_{2m} =$ "not $B(y)$"; otherwise set $A_{2m} = A_{2m-1}$.

We prove several claims about the set of formulas $\mathscr{H} = \{A_1, A_2, \ldots\}$ which will correspond to the fact that the sequence describes a certain structure $\mathscr{S}$ as desired.

(3) For all n, A is not provable from $\{A_1, \ldots, A_n\}$.
Proof. This follows immediately from the construction and properties (1) and (2). *qed*(3)

(4) For every formula B, "not B" $\in \mathscr{H}$ iff $B \notin \mathscr{H}$.
Proof. Consider $B = F_m$. Assume that "not B" $\in \mathscr{H}$. Assume for a contradiction that also $B \in \mathscr{H}$. Choose a natural number n such that B, "not B"$\in \{A_1, \ldots, A_n\}$. Then A is immediately provable from $\{A_1, \ldots, A_n\}$ by the rules of *contradiction* and *ex falsum libenter*. But this contradicts (3).

Conversely assume that “not B” $\notin \mathscr{H}$. Then by construction of $\mathscr{H}$, $A_{2m-1} = F_m = B \in \mathscr{H}$. *qed*(4)

(5) Let B be provable from $\mathscr{H}$. Then $B \in \mathscr{H}$.
Proof. Let Proof1, B be a proof of B from $\mathscr{H}$. Assume $B \notin \mathscr{H}$. By (4), “not B” $\in \mathscr{H}$. Then the following text is a proof of A from $\mathscr{H}$:

k	Statement	Rule ... with hypothesis ...
1	Proof1	given
2	B.	given
3	not B.	copying “not B” out of $\mathscr{H}$
4	Contradiction.	*contradiction* with 2, 3
5	A.	*ex falsum libenter* with 4

This contradicts (3). *qed*(5).

(6) “not A”$\in \mathscr{H}$.
Proof. By (3), $A \notin \mathscr{H}$. The claim follows by (4). *qed*(6)

(7) “contradiction”$\notin \mathscr{H}$.
Proof. If “contradiction”$\in \mathscr{H}$, say “contradiction”$= A_n$ then A is provable from $\{A_1, \ldots, A_n\}$ by the *ex falsum libenter* rule, which contradicts (3). *qed*(7)

(8) For all formulas B and C, we have “B implies C”$\in \mathscr{H}$ iff ($B \in \mathscr{H}$ implies $C \in \mathscr{H}$).
Proof. Assume “B implies C”$\in \mathscr{H}$ and assume that $B \in \mathscr{H}$. Then C is provable from $\mathscr{H}$. By (5), $C \in \mathscr{H}$, and thus $B \in \mathscr{H}$ implies $C \in \mathscr{H}$.

Conversely assume that “B implies C”$\notin \mathscr{H}$. By (4), “not (B implies C)”$\in \mathscr{H}$. From “not (B implies C)” one can prove B and “not C”. By (5), $B \in \mathscr{H}$ and $C \notin \mathscr{H}$. Hence $B \in \mathscr{H}$ does not imply $C \in \mathscr{H}$. *qed*(8)

(9) For all formulas $B(x)$ we have: “for all x holds $B(x)$”$\in \mathscr{H}$ iff for all variables y holds $B(y) \in \mathscr{H}$.
Proof. Assume that “for all x holds $B(x)$”$\in \mathscr{H}$. Then for all variables y, $B(y)$ is provable from $\mathscr{H}$ by the rule of *instantiation*. By (5), $B(y) \in \mathscr{H}$.

Conversely assume that “for all x holds $B(x)$”$\notin \mathscr{H}$. By (4), “not for all x holds $B(x)$”$\in \mathscr{H}$. Choose an index m such that $F_m =$“not for all x holds $B(x)$”. By construction, $A_{2m-1} = F_m$ and $A_{2m} =$“not $B(y)$”$\in \mathscr{H}$ for some variable y. By (4), $B(y) \notin \mathscr{H}$. *qed*(9)

Now define the structure $\mathscr{S} = (S, \ldots)$ as follows. Let S be the set of all variables occurring in $A_0, A_1, \ldots$. For every n-ary relation symbol R occurring in $A_0, A_1, \ldots$ define an n-ary relation R^S on S by

$$R^S(x_1,\ldots,x_n) \text{ iff } R(x_1,\ldots,x_n) \in \mathscr{H}.$$

(10) Let F be a formula. Then F holds in $\mathscr{S}$ iff $F \in \mathscr{H}$.
Proof. We prove the claim by induction on the length of F as a sequence of symbols. So assume that the claim holds for all shorter F'.
Case 1. F is a relational formula of the form $F = R(x_1,\ldots,x_n)$.
Then by definition of the structure $\mathscr{S}$, F holds in $\mathscr{S}$ iff $R^S(x_1,\ldots,x_n)$ iff $R(x_1,\ldots,x_n) \in \mathscr{H}$.
Case 2. $F =$"contradiction".
Then F does *not* hold in $\mathscr{S}$. Also, by (7), $F \notin \mathscr{H}$.
Case 3. $F =$ "not B".
Then by the inductive assumption and (4),
F holds in $\mathscr{S}$
iff B does not hold in $\mathscr{S}$
iff $B \notin \mathscr{H}$
iff "not B" $\in \mathscr{H}$.
Case 4. $F =$"B implies C".
Then by the inductive assumption and (8),
"B implies C" holds in $\mathscr{S}$
iff B holds in $\mathscr{S}$ implies C holds in $\mathscr{S}$
iff $B \in \mathscr{H}$ implies $C \in \mathscr{H}$
iff "B implies C"$\in \mathscr{H}$.
Case 5. $F =$"for all x holds $B(x)$".
Then by the inductive assumption and (9),
"for all x holds $B(x)$" holds in $\mathscr{S}$
iff for all variables $y \in S$, $B(y)$ holds in $\mathscr{S}$
iff for all variables $y \in S$, $B(y) \in \mathscr{H}$
iff "for all x holds $B(x)$"$\in \mathscr{H}$. *qed*(10)

By (10) and (6), the initial formula A does not hold in $\mathscr{S}$. Thus A is not universally valid. □

6 GENERAL REMARKS

The GÖDEL *completeness* theorem is sometimes considered to be of lesser importance than the far reaching *incompleteness* theorems. Technically, however, the completeness theorem is essential for the incompleteness theorem, since it yields distinguished, complete calculi for which one then can examine incompleteness and consistency questions. Moreover, due to the effective nature of the proof calculi, the HENKIN-style model construction, and its universal applicability to all first-order theories the completeness theorem has a broad spectrum of "positive" applications which may be even wider than the limitative consequences of the incompleteness theorem. We list a few application areas.

Mathematical applications: The completeness theorem and its proofs have immediate consequences in first-order logic like the compactness theorem or the LÖWENHEIM-SKOLEM theorems, which can be used to construct structures with particular or unusual properties. This has many algebraic and other applications, including nonstandard analysis.

Mathematical correctness and proof checking: The completeness theorem provides an absolute *criterion for the correctness of proofs*. A mathematical proof is correct if and only if it can (in principle) be reformulated as a formal proof. Although one usually carries out informal or "semi-formal" proofs, one may if there is any doubt formalize and atomize arguments in such a way that proofs or parts thereof become fully formal and may also be checked by computer (*automatic proof checking*).

Formalization in other areas: The success of the formal method in mathematics has provided motivations for other sciences to formalize their statements and methods as far as possible. This corresponds to the tendency to present the world by data which can be operated on by algorithms.

Automatic theorem proving. In principle all universally valid statements A can be proved automatically: enumerate all possible (proof) texts and check automatically whether the enumeration contains a proof of A. The success of this method is guaranteed by the completeness theorem. In general this method is not feasable due to the enormeous size of the set in which the proofs are searched for. But for limited domains there are now practical automatic theorem provers.

Artificial intelligence: The beginnings of artificial intelligence were characterized by the logical formalization of the state of the environment in question and by applying automatic proving techniques to answer questions about the environment. Due to the complexity problems mentioned above, this approach is meanwhile considered to be unrealistic.

Non-classical logics. First-order predicate logic is considered to be an optimal logic in the sense that it has a complete, decidable proof calculus which connects syntax and semantics. Many other languages and logics have been designed and studied on the pattern of first-order logic: modal logics, temporal logics, algorithmic logics.

REFERENCES

[1] Gerhard Gentzen. Untersuchungen über das logische Schließen. I and II. *Mathematische Zeitschrift*, 39:176–210 and 405–431, 1935.

[2] Kurt Gödel. Über die Vollständigkeit der Axiome des logischen Funktionenkalküls. *Monatshefte für Mathematik und Physik*, 36:349–360, 1930.

[3] Kurt Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme. *Monatshefte für Mathematik und Physik*, 38:173–198, 1931.

[4] Gisbert Hasenjaeger. Eine Bemerkung zu Henkin's Beweis für die Vollständigkeit des Prädikatenkalküls der ersten Stufe. *Journal of Symbolic Logic*, 18:42–48, 1953.

[5] Leon Henkin. Completeness in the theory of types. *Journal of Symbolic Logic*, 15(2):81–91, 1950.

[6] Peter Koepke and Bernhard Schröder. Natürlich formal. In B. Willee et al, editors, *Computational Linguistics — Achievements and Perspectives*. Gardez!-Verlag, 2003.

[7] `http://www.math.uni-bonn.de/people/naproche`