The first-order theory of canonical calculi ($\mathbf{CC}^*$) and its language $\mathcal{L}^{1*}$

András Máté

18.10.2024

The goal of **CC***

The goal of $\mathbf{CC}^*$

$\mathbf{CC}^*$ is the rewriting of the (hyper)calculus $\mathbf{H}_3$ in the form of a first-order theory. (See 27th September presentation.)
$\mathbf{H}_3$ derives strings like Ka, Wb, aDb, aGb, Aa with the intended meanings 'a is a calculus', …'a is an autonomous number'. We want $\mathbf{CC}^*$ to prove formulas like $K(a), \ldots A(a)$ just in the same case.

The goal of **CC***

CC* is the rewriting of the (hyper)calculus $\mathbf{H}_3$ in the form of a first-order theory. (See 27th September presentation.)
$\mathbf{H}_3$ derives strings like Ka, Wb, aDb, aGb, Aa with the intended meanings 'a is a calculus', …'a is an autonomous number'. We want **CC*** to prove formulas like $K(a), \dots A(a)$ just in the same case.

The language of **CC*** is the first-order language $\mathcal{L}^{1*}$.
Non-logical components :

The goal of **CC***

CC* is the rewriting of the (hyper)calculus $\mathbf{H}_3$ in the form of a first-order theory. (See 27th September presentation.)
$\mathbf{H}_3$ derives strings like Ka, Wb, aDb, aGb, Aa with the intended meanings 'a is a calculus', ...'a is an autonomous number'. We want **CC*** to prove formulas like $K(a), \ldots A(a)$ just in the same case.

The language of **CC*** is the first-order language $\mathcal{L}^{1*}$.
Non-logical components :

- $N_\varnothing = \{\vartheta,\ \alpha,\ \beta,\ \xi,\ \gg,\ *\}$
  ϑ denotes the empty string, the other name constants denote (autonymously) the letters of $\mathcal{A}_{cc}$.

The goal of **CC***

CC* is the rewriting of the (hyper)calculus $\mathbf{H}_3$ in the form of a first-order theory. (See 27th September presentation.)
$\mathbf{H}_3$ derives strings like Ka, Wb, aDb, aGb, Aa with the intended meanings 'a is a calculus', ... 'a is an autonomous number'. We want **CC*** to prove formulas like $K(a), \ldots A(a)$ just in the same case.

The language of **CC*** is the first-order language $\mathcal{L}^{1*}$.
Non-logical components :

- $N_\varnothing = \{\vartheta,\ \alpha,\ \beta,\ \xi,\ \gg,\ *\}$
  ϑ denotes the empty string, the other name constants denote (autonymously) the letters of $\mathcal{A}_{cc}$.
- $N_{oo} = \{\varnothing\}$
  The empty string denotes concatenation (and we omit the parentheses around its arguments), i.e., we write the concatenation of the strings x and y as xy.

The language $\mathcal{L}^{1*}$ (continuation)

The auxiliary letters of the hypercalculi $\mathbf{H}_1 - \mathbf{H}_3$ become predicates and we keep the intended meanings:

The language $\mathcal{L}^{1*}$ (continuation)

The auxiliary letters of the hypercalculi $\mathbf{H}_1$ – $\mathbf{H}_3$ become predicates and we keep the intended meanings:

- $P_o = \{I,\ L,\ V,\ W,\ T,\ R,\ K,\ A\}$

The language $\mathcal{L}^{1*}$ (continuation)

The auxiliary letters of the hypercalculi $\mathbf{H}_1$ – $\mathbf{H}_3$ become predicates and we keep the intended meanings:

- $P_o = \{I,\ L,\ V,\ W,\ T,\ R,\ K,\ A\}$
- $P_{oo} = \{D,\ F,\ G\}$

The language $\mathcal{L}^{1*}$ (continuation)

The auxiliary letters of the hypercalculi $\mathbf{H}_1$ – $\mathbf{H}_3$ become predicates and we keep the intended meanings:

- $P_o = \{I,\ L,\ V,\ W,\ T,\ R,\ K,\ A\}$
- $P_{oo} = \{D,\ F,\ G\}$
- $P_{oooo} = \{S\}$
  $S(v)(u)(y)(x)$: if we substitute the word y for the variable x, we get the string v from the string u.)

The language $\mathcal{L}^{1*}$ (continuation)

The auxiliary letters of the hypercalculi $\mathbf{H}_1 - \mathbf{H}_3$ become predicates and we keep the intended meanings:

- $P_o = \{I,\ L,\ V,\ W,\ T,\ R,\ K,\ A\}$
- $P_{oo} = \{D,\ F,\ G\}$
- $P_{oooo} = \{S\}$
  $S(v)(u)(y)(x)$: if we substitute the word y for the variable x, we get the string v from the string u.)

Logical constants, variables (let us write them as $\mathfrak{x}, \mathfrak{x}_1, \ldots$), the syntax of terms and formulas are like in any other first-order language. The intended universe (the domain of the variables) is the class of $\mathcal{A}_{cc}$-strings.

The axioms of $\mathbf{CC}^*$: the language radix-axioms

First group: The $\mathcal{A}_{cc}$-strings build a language radix (13th September presentation) or in the language of algebra, the free monoid on $\mathcal{A}_{cc}$. In some details:

The axioms of $\mathbf{CC}^*$: the language radix-axioms

First group: The $\mathcal{A}_{cc}$-strings build a language radix (13th September presentation) or in the language of algebra, the free monoid on $\mathcal{A}_{cc}$. In some details:

- The empty string is different from the letters (five axioms).

The axioms of **CC***: the language radix-axioms

First group: The $\mathcal{A}_{cc}$-strings build a language radix (13th September presentation) or in the language of algebra, the free monoid on $\mathcal{A}_{cc}$. In some details:

- The empty string is different from the letters (five axioms).
- Strings ending with different letters are different (ten axioms).

The axioms of $\mathbf{CC}^*$: the language radix-axioms

First group: The $\mathcal{A}_{cc}$-strings build a language radix (13th September presentation) or in the language of algebra, the free monoid on $\mathcal{A}_{cc}$. In some details:

- The empty string is different from the letters (five axioms).
- Strings ending with different letters are different (ten axioms).
- Five more axioms about strings:
  1. $\forall\mathfrak{x}(\mathfrak{x}\vartheta = \vartheta\mathfrak{x} = \mathfrak{x})$
  2. $\forall\mathfrak{x}\forall\mathfrak{x}_1(\mathfrak{x}\mathfrak{x}_1 = \vartheta \supset (\mathfrak{x} = \vartheta \wedge \mathfrak{x}_1 = \vartheta)$
  3. $\forall\mathfrak{x}_1\exists\mathfrak{x}(\mathfrak{x}_1 \neq \vartheta \supset (\mathfrak{x}_1 = \mathfrak{x}\alpha \vee \mathfrak{x}_1 = \mathfrak{x}\beta \vee \mathfrak{x}_1 = \mathfrak{x}\xi \vee \mathfrak{x}_1 = \mathfrak{x} \gg$ $\vee\mathfrak{x}_1 = \mathfrak{x}*))$
  4. $\forall\mathfrak{x}\forall\mathfrak{x}_1\forall\mathfrak{x}_2(\mathfrak{x}_1\mathfrak{x} = \mathfrak{x}_2\mathfrak{x} \supset \mathfrak{x}_1 = \mathfrak{x}_2)$

The axioms of $\mathbf{CC}^*$: the language radix-axioms

First group: The $\mathcal{A}_{cc}$-strings build a language radix (13th September presentation) or in the language of algebra, the free monoid on $\mathcal{A}_{cc}$. In some details:

- The empty string is different from the letters (five axioms).
- Strings ending with different letters are different (ten axioms).
- Five more axioms about strings:
    1. $\forall\mathfrak{x}(\mathfrak{x}\vartheta = \vartheta\mathfrak{x} = \mathfrak{x})$
    2. $\forall\mathfrak{x}\forall\mathfrak{x}_1(\mathfrak{x}\mathfrak{x}_1 = \vartheta \supset (\mathfrak{x} = \vartheta \wedge \mathfrak{x}_1 = \vartheta)$
    3. $\forall\mathfrak{x}_1\exists\mathfrak{x}(\mathfrak{x}_1 \neq \vartheta \supset (\mathfrak{x}_1 = \mathfrak{x}\alpha \vee \mathfrak{x}_1 = \mathfrak{x}\beta \vee \mathfrak{x}_1 = \mathfrak{x}\xi \vee \mathfrak{x}_1 = \mathfrak{x} \gg \vee\mathfrak{x}_1 = \mathfrak{x}*))$
    4. $\forall\mathfrak{x}\forall\mathfrak{x}_1\forall\mathfrak{x}_2(\mathfrak{x}_1\mathfrak{x} = \mathfrak{x}_2\mathfrak{x} \supset \mathfrak{x}_1 = \mathfrak{x}_2)$

The axioms of $\mathbf{CC}^*$: the language radix-axioms

First group: The $\mathcal{A}_{cc}$-strings build a language radix (13th September presentation) or in the language of algebra, the free monoid on $\mathcal{A}_{cc}$. In some details:

- The empty string is different from the letters (five axioms).
- Strings ending with different letters are different (ten axioms).
- Five more axioms about strings:
  1. $\forall\mathfrak{x}(\mathfrak{x}\vartheta = \vartheta\mathfrak{x} = \mathfrak{x})$
  2. $\forall\mathfrak{x}\forall\mathfrak{x}_1(\mathfrak{x}\mathfrak{x}_1 = \vartheta \supset (\mathfrak{x} = \vartheta \wedge \mathfrak{x}_1 = \vartheta)$
  3. $\forall\mathfrak{x}_1\exists\mathfrak{x}(\mathfrak{x}_1 \neq \vartheta \supset (\mathfrak{x}_1 = \mathfrak{x}\alpha \vee \mathfrak{x}_1 = \mathfrak{x}\beta \vee \mathfrak{x}_1 = \mathfrak{x}\xi \vee \mathfrak{x}_1 = \mathfrak{x} \gg$ $\vee\mathfrak{x}_1 = \mathfrak{x}*))$
  4. $\forall\mathfrak{x}\forall\mathfrak{x}_1\forall\mathfrak{x}_2(\mathfrak{x}_1\mathfrak{x} = \mathfrak{x}_2\mathfrak{x} \supset \mathfrak{x}_1 = \mathfrak{x}_2)$

Remark: The textbook defines the theorems of $\mathbf{CC}^*$ by a canonical calculus Σ^*. We omit this step; but you can find the axioms of this slide as rules 61-80. of Σ^* on p. 80. of the textbook. The notation is a little bit different.

CC*'s calculus axioms (continuation)

CC*'s calculus axioms (continuation)

- The arrows are substituted by the conditional sign '$\supset$' and formulas of the form $\ulcorner A \supset B \supset C \urcorner$ are understood as $\ulcorner (A \supset (B \supset C)) \urcorner$.

CC*'s calculus axioms (continuation)

- The arrows are substituted by the conditional sign '$\supset$' and formulas of the form $\ulcorner A \supset B \supset C \urcorner$ are understood as $\ulcorner (A \supset (B \supset C)) \urcorner$.
- The open formulas obtained by the previous rules are substituted by their universal closure.

$\mathbf{CC}^*$'s calculus axioms (continuation)

- The arrows are substituted by the conditional sign '$\supset$' and formulas of the form $\ulcorner A \supset B \supset C \urcorner$ are understood as $\ulcorner (A \supset (B \supset C)) \urcorner$.
- The open formulas obtained by the previous rules are substituted by their universal closure.

The axioms of $\mathbf{CC}^*$ are the 20 language radix-axioms plus the 34 axioms obtained from the rules of $\mathbf{H}_3$. E.g., the rules 12. and 13. of $\mathbf{H}_3$ (defining the extension of K) become the following axioms:

- $\forall \mathfrak{x}(R(\mathfrak{x}) \supset K(\mathfrak{x}))$
- $\forall \mathfrak{x} \forall \mathfrak{x}_1(K(\mathfrak{x}) \supset R(\mathfrak{x}_1) \supset K(\mathfrak{x} * \mathfrak{x}_1))$

$\mathbf{CC}^*$'s calculus axioms (continuation)

- The arrows are substituted by the conditional sign '$\supset$' and formulas of the form $\ulcorner A \supset B \supset C \urcorner$ are understood as $\ulcorner (A \supset (B \supset C)) \urcorner$.
- The open formulas obtained by the previous rules are substituted by their universal closure.

The axioms of $\mathbf{CC}^*$ are the 20 language radix-axioms plus the 34 axioms obtained from the rules of $\mathbf{H}_3$. E.g., the rules 12. and 13. of $\mathbf{H}_3$ (defining the extension of K) become the following axioms:

- $\forall \mathfrak{x}(R(\mathfrak{x}) \supset K(\mathfrak{x}))$
- $\forall \mathfrak{x} \forall \mathfrak{x}_1 (K(\mathfrak{x}) \supset R(\mathfrak{x}_1) \supset K(\mathfrak{x} * \mathfrak{x}_1))$

The above rules of translation apply to any string derivable in $\mathbf{H}_3$. Let us denote the translation on the string f into a $\mathcal{L}^{1*}$-formula $Tr(f)$.

Denotation and truth in $\mathcal{L}^{1*}$

Denotation and truth in $\mathcal{L}^{1*}$

We can give a truth definition for $\mathcal{L}^{1*}$-formulas, independently from any set-theoretical semantics.

Denotation and truth in $\mathcal{L}^{1*}$

We can give a truth definition for $\mathcal{L}^{1*}$-formulas, independently from any set-theoretical semantics.

The closed terms of $\mathcal{L}^{1*}$ are $\mathcal{A}_{cc} \cup \{\vartheta\}$-strings. They denote $\mathcal{A}_{cc}$-strings that we obtain from them by deleting the ϑs.

Denotation and truth in $\mathcal{L}^{1*}$

We can give a truth definition for $\mathcal{L}^{1*}$-formulas, independently from any set-theoretical semantics.

The closed terms of $\mathcal{L}^{1*}$ are $\mathcal{A}_{cc} \cup \{\vartheta\}$-strings. They denote $\mathcal{A}_{cc}$-strings that we obtain from them by deleting the ϑs.

- A closed atomic formula $\ulcorner s = t \urcorner$ is true iff 's' and 't' denote the same string.

Denotation and truth in $\mathcal{L}^{1*}$

We can give a truth definition for $\mathcal{L}^{1*}$-formulas, independently from any set-theoretical semantics.

The closed terms of $\mathcal{L}^{1*}$ are $\mathcal{A}_{cc} \cup \{\vartheta\}$-strings. They denote $\mathcal{A}_{cc}$-strings that we obtain from them by deleting the ϑs.

- A closed atomic formula $\ulcorner s = t \urcorner$ is true iff 's' and 't' denote the same string.
- Closed atomic formulas containing the predicates $I,\ L,\ W,\ V,\ T,\ R,\ K,\ F,\ S$ are true iff they are true according to the intended interpretation. I.e., $\ulcorner I(s) \urcorner$ is true iff the string s is an index, $\ulcorner K(s) \urcorner$ is true iff s is a code of a calculus, $\ulcorner S(s)(t)(v)(u) \urcorner$ is true iff by substituting the word (variable-free string) v for the variable u in the string t, we get s, etc.

Denotation and truth in $\mathcal{L}^{1*}$

We can give a truth definition for $\mathcal{L}^{1*}$-formulas, independently from any set-theoretical semantics.

The closed terms of $\mathcal{L}^{1*}$ are $\mathcal{A}_{cc} \cup \{\vartheta\}$-strings. They denote $\mathcal{A}_{cc}$-strings that we obtain from them by deleting the ϑs.

- A closed atomic formula $\ulcorner s = t \urcorner$ is true iff 's' and 't' denote the same string.
- Closed atomic formulas containing the predicates I, L, W, V, T, R, K, F, S are true iff they are true according to the intended interpretation. I.e., $\ulcorner I(s) \urcorner$ is true iff the string s is an index, $\ulcorner K(s) \urcorner$ is true iff s is a code of a calculus, $\ulcorner S(s)(t)(v)(u) \urcorner$ is true iff by substituting the word (variable-free string) v for the variable u in the string t, we get s, etc.

These two stipulations are effective, so the reference to the intended interpretation is not problematic.

Truth definition (continuation)

Truth definition (continuation)

- $\ulcorner D(s)(t) \urcorner$ is true iff $H_3 \mapsto s' D t'$ where s' and t' are the strings denoted by s resp. t, i.e. the calculus encoded by s derives the string t.

Truth definition (continuation)

- $\ulcorner D(s)(t) \urcorner$ is true iff $H_3 \mapsto s'Dt'$ where s' and t' are the strings denoted by s resp. t, i.e. the calculus encoded by s derives the string t.
- $\ulcorner A(s) \urcorner$ is true iff $H_3 \mapsto As'$, i. e. s is an autonomous numeral.

Truth definition (continuation)

- $\ulcorner D(s)(t) \urcorner$ is true iff $H_3 \mapsto s'Dt'$ where s' and t' are the strings denoted by s resp. t, i.e. the calculus encoded by s derives the string t.
- $\ulcorner A(s) \urcorner$ is true iff $H_3 \mapsto As'$, i. e. s is an autonomous numeral.

We could use the method of these stipulations at the first two items, too. But these last ones are not effective (just this is our point).

Truth definition (continuation)

- $\ulcorner D(s)(t) \urcorner$ is true iff $H_3 \mapsto s'Dt'$ where s' and t' are the strings denoted by s resp. t, i.e. the calculus encoded by s derives the string t.
- $\ulcorner A(s) \urcorner$ is true iff $H_3 \mapsto As'$, i. e. s is an autonomous numeral.

We could use the method of these stipulations at the first two items, too. But these last ones are not effective (just this is our point).

- The evaluation of negation and conditional goes as usual.

Truth definition (continuation)

- $\ulcorner D(s)(t) \urcorner$ is true iff $H_3 \mapsto s' D t'$ where s' and t' are the strings denoted by s resp. t, i.e. the calculus encoded by s derives the string t.
- $\ulcorner A(s) \urcorner$ is true iff $H_3 \mapsto As'$, i. e. s is an autonomous numeral.

We could use the method of these stipulations at the first two items, too. But these last ones are not effective (just this is our point).

- The evaluation of negation and conditional goes as usual.
- Truth condition for universal quantification is given by *substitution* (that's why we don't need set theory): $\ulcorner \forall x A \urcorner$ is false iff for some t $\mathcal{A}_{cc}$-string, $[A]^{t/x}$ is false, and true in the other case. (Remark: x is a metalanguage variable here running on the $\mathcal{L}^{1*}$-variables $\mathfrak{x}_n$.)

Truth definition (continuation)

- $\ulcorner D(s)(t) \urcorner$ is true iff $H_3 \mapsto s'Dt'$ where s' and t' are the strings denoted by s resp. t, i.e. the calculus encoded by s derives the string t.
- $\ulcorner A(s) \urcorner$ is true iff $H_3 \mapsto As'$, i. e. s is an autonomous numeral.

We could use the method of these stipulations at the first two items, too. But these last ones are not effective (just this is our point).

- The evaluation of negation and conditional goes as usual.
- Truth condition for universal quantification is given by *substitution* (that's why we don't need set theory): $\ulcorner \forall x A \urcorner$ is false iff for some t $\mathcal{A}_{cc}$-string, $[A]^{t/x}$ is false, and true in the other case. (Remark: x is a metalanguage variable here running on the $\mathcal{L}^{1*}$-variables $\mathfrak{x}_n$.)
- Open formulas of $\mathcal{L}^{1*}$ are true iff their universal closure is true.

The consistency of **CC***

The consistency of $\mathbf{CC}^*$

Theorem: All the theorems of $\mathbf{CC}^*$ are true according to the above truth definition.

Theorem: All the theorems of $\mathbf{CC}^*$ are true according to the above truth definition.

The proof goes by induction following the inductive definition of $\Gamma \vdash A$ (previous presentation): the axioms of $\mathbf{CC}^*$ are true (simple calculation), the basic formulas of first-order logic (of $\mathcal{L}^{1*}$) are true and detachment preserves truth.

The consistency of $\mathbf{CC}^*$

Theorem: All the theorems of $\mathbf{CC}^*$ are true according to the above truth definition.

The proof goes by induction following the inductive definition of $\Gamma \vdash A$ (previous presentation): the axioms of $\mathbf{CC}^*$ are true (simple calculation), the basic formulas of first-order logic (of $\mathcal{L}^{1*}$) are true and detachment preserves truth.

Corollary: $\mathbf{CC}^*$ is consistent. Because there are false sentences of $\mathcal{L}^{1*}$ (e.g., '$\alpha = \beta$'), and according to the theorem, they are not provable.

Theorem: All the theorems of $\mathbf{CC}^*$ are true according to the above truth definition.

The proof goes by induction following the inductive definition of $\Gamma \vdash A$ (previous presentation): the axioms of $\mathbf{CC}^*$ are true (simple calculation), the basic formulas of first-order logic (of $\mathcal{L}^{1*}$) are true and detachment preserves truth.

Corollary: $\mathbf{CC}^*$ is consistent. Because there are false sentences of $\mathcal{L}^{1*}$ (e.g., '$\alpha = \beta$'), and according to the theorem, they are not provable.

Theorem: If $\mathbf{H}_3 \mapsto f$, then $Tr(f)$ is provable in $\mathbf{CC}^*$.
The proof goes by induction following the inductive definition of strings derivable in $\mathbf{H}_3$.

Undecidability

Theorem: $\mathbf{CC}^*$ is not decidable.

Suppose we have an algorithm to decide which sentences of $\mathcal{L}^{1*}$ are theorems of $\mathbf{CC}^*$. In this case, we could decide which sentences of the form $A(c)$ (where c is a numeral) are theorems. But this would mean that we could decide which numerals are autonomous - in contradiction to our earlier result that the class of autonomous numerals is not decidable.

Theorem: **CC*** is not decidable.
Suppose we have an algorithm to decide which sentences of $\mathcal{L}^{1*}$ are theorems of **CC***. In this case, we could decide which sentences of the form $A(c)$ (where c is a numeral) are theorems. But this would mean that we could decide which numerals are autonomous - in contradiction to our earlier result that the class of autonomous numerals is not decidable.

Theorem(Church-Turing-Markov): First-order logic is not decidable.
I. e., there is no algorithm for every first-order language that decides about every formula whether it is a logical truth (consequence of the empty set of formulas) or not.
E.g., for $\mathcal{L}^{1*}$ there is no such algorithm. Because otherwise we had an algorithm to decide which formulas of the form $\mathbf{Ax} \supset A(c)$ are logical truths (where **Ax** is the conjunction of all axioms of **CC*** and c is a numeral). This would imply the decidability of the class of autonomous numerals again.

Negation completeness

Negation completeness

A theory is omniscient (knows everything about its subject) if it can decide every question that can be formulated in the language of the theory.

Negation completeness

A theory is omniscient (knows everything about its subject) if it can decide every question that can be formulated in the language of the theory.

Formal counterpart: a formal theory is negation complete iff for every sentence A of its language, either A or $\neg A$ is a theorem. Otherwise, it's called (negation) incomplete.

Negation completeness

A theory is omniscient (knows everything about its subject) if it can decide every question that can be formulated in the language of the theory.

Formal counterpart: a formal theory is negation complete iff for every sentence A of its language, either A or $\neg A$ is a theorem. Otherwise, it's called (negation) incomplete.

Inconsistent theories are negation complete but uninteresting.

Negation completeness

A theory is omniscient (knows everything about its subject) if it can decide every question that can be formulated in the language of the theory.

Formal counterpart: a formal theory is negation complete iff for every sentence A of its language, either A or $\neg A$ is a theorem. Otherwise, it's called (negation) incomplete.

Inconsistent theories are negation complete but uninteresting.

A plausible idea: if an axiomatic theory proves to be negation incomplete, then it was too weak. Let us extend the set of axioms by some true sentences.

Negation completeness

A theory is omniscient (knows everything about its subject) if it can decide every question that can be formulated in the language of the theory.

Formal counterpart: a formal theory is negation complete iff for every sentence A of its language, either A or $\neg A$ is a theorem. Otherwise, it's called (negation) incomplete.

Inconsistent theories are negation complete but uninteresting.

A plausible idea: if an axiomatic theory proves to be negation incomplete, then it was too weak. Let us extend the set of axioms by some true sentences.

There are very weak but negation complete theories, e.g. the theory of two-member world.

Negation completeness

A theory is omniscient (knows everything about its subject) if it can decide every question that can be formulated in the language of the theory.

Formal counterpart: a formal theory is negation complete iff for every sentence A of its language, either A or $\neg A$ is a theorem. Otherwise, it's called (negation) incomplete.

Inconsistent theories are negation complete but uninteresting.

A plausible idea: if an axiomatic theory proves to be negation incomplete, then it was too weak. Let us extend the set of axioms by some true sentences.

There are very weak but negation complete theories, e.g. the theory of two-member world.

The interesting case is when a theory is incomplete because it is too strong, and therefore the incompleteness cannot be remedied by extending the theory.

The theory **CC**

CC comes from the theory $\mathbf{CC}^*$ by deleting some predicates from the language $\mathcal{L}^{1*}$ and the axioms belonging to them from the axioms and adding one more auxiliary axiom called SUD (to be specified at the next class).

The theory **CC**

CC comes from the theory $\mathbf{CC}^*$ by deleting some predicates from the language $\mathcal{L}^{1*}$ and the axioms belonging to them from the axioms and adding one more auxiliary axiom called SUD (to be specified at the next class).

In some details:
The language $\mathcal{L}^{10}$ of **CC** is the same as $\mathcal{L}^{1*}$ except of that it does not contain the two-place predicates F and G and the one-place predicate A.

The theory **CC**

CC comes from the theory $\mathbf{CC}^*$ by deleting some predicates from the language $\mathcal{L}^{1*}$ and the axioms belonging to them from the axioms and adding one more auxiliary axiom called SUD (to be specified at the next class).

In some details:
The language $\mathcal{L}^{10}$ of **CC** is the same as $\mathcal{L}^{1*}$ except of that it does not contain the two-place predicates F and G and the one-place predicate A.

The class of axioms Γ_0 of **CC** comes from the axioms of $\mathbf{CC}^*$ by omitting the last nine axioms corresponding the rules 26.-34. of $\mathbf{H}_3$ (i.e, it contains the axioms that translate the rules of $\mathbf{H}_2$ but not the further rules of $\mathbf{H}_3$ governing the predicates omitted) and by adding SUD.

The theory **CC**

CC comes from the theory $\mathbf{CC}^*$ by deleting some predicates from the language $\mathcal{L}^{1*}$ and the axioms belonging to them from the axioms and adding one more auxiliary axiom called SUD (to be specified at the next class).

In some details:
The language $\mathcal{L}^{10}$ of **CC** is the same as $\mathcal{L}^{1*}$ except of that it does not contain the two-place predicates F and G and the one-place predicate A.

The class of axioms Γ_0 of **CC** comes from the axioms of $\mathbf{CC}^*$ by omitting the last nine axioms corresponding the rules 26.-34. of $\mathbf{H}_3$ (i.e, it contains the axioms that translate the rules of $\mathbf{H}_2$ but not the further rules of $\mathbf{H}_3$ governing the predicates omitted) and by adding SUD.

We can apply the truth definition we have specified at the last class. We will show that SUD is true according to this definition, too. Therefore, the theorems of **CC** are all true and the theory is consistent.