Math. Struct. in Comp. Science (2002), *vol.* 12, *pp.* 1–18.
DOI: 10.1017/S0960129501003449 Printed in the United Kingdom

Syntactical truth predicates for formulas with atomic negation

LOIC COLSON[†]

University of Metz - L.I.T.A.
Département d'Informatique, UFR MIM, Ile du Saulcy 57000 Metz - France.

Email: colson@lita.univ-metz.fr

Received 6 April 1999; revised 7 February 2001

We define a language for second-order arithmetic using the (positive) connectives $\dot\wedge$ and $\dot\vee$, the (positive) quantifiers $\dot\forall$ and $\dot\exists$, and with negation 'pushed to the atoms' under the form of atomic formula $t \dot= u, t \dot\neq u, t \dot\in X$, $t \dot\notin X$ where t and u stand for first-order terms and X for a second-order variable.
We define and study the translation from this language to the more usual *implicational* language containing the entailment connective $\rightarrow$. We also study the converse translation. Next we define the appropriate notion of *syntactical truth predicate* for this language (such a notion has been introduced for the implicational language in Colson and Grigorieff (1998)). We establish using the previous translations that the existence of such a predicate for this language is equivalent to the the existence of such a predicate for the implicational language. This result is established in a predicative formal system. We conclude by discussing some elementary attempts to construct such a truth predicate by a fixed point technique.

1. Introduction

1.1. *Syntactical truth predicates for implicational formulas*

Syntactical truth predicates have been introduced in (Colson and Grigorieff 1998) as a proposal for a predicative approach to the semantics of impredicative second-order arithmetic $\boldsymbol{PA^2}$. The formulas of $\boldsymbol{PA^2}$ (formalized *à la Takeuti* (Takeuti 1975)) that were considered in Colson and Grigorieff (1998) were made of *equality statements* $t = u$ (where t and u are terms of some first-order language standing for natural numbers), *membership statements* $t \in X$ (where X is a second-order variable standing for a set of natural numbers), entailment statements $A \Rightarrow B$ (where A and B are two previously constructed formulas) and first or second-order universal quantifications $\forall xA$ and $\forall XA$. First-order substitution $A[x \leftarrow t]$ of the term t for the first-order variable x in the formula A is defined in the natural way by induction on A. Second-order substitution needs a little preliminary definition: a *set definition* is an expression like λxB (where B is a formula) and means

[†] This research was partly supported by Chalmers University of Technology and Université de Marne-la-Vallée

'the set of numbers x such that $B(x)$ holds'. Here we use Church's lambda-notation for the propositional function mapping the natural number x to the proposition $B(x)$. The substitution $A[X \leftarrow D]$ of the set definition D for the variable X in the formula A is then defined by induction on A. The critical case is $(t \in X)[X \leftarrow \lambda x B]$ defined as $B[x \leftarrow t]$. Here we shall call these formulas *implicational formulas*.

The notion of syntactical truth predicate for this language, which we will call in the present paper *syntactical truth predicate for implicational formulas*, can then be introduced as a set T of closed implicational formulas such that

(*i*) $T(t = u)$ iff the closed terms t and u are convertible, that is, have the same value in the standard interpretation
(*ii*) $T(A \Rightarrow B)$ iff $(T(A) \Rightarrow T(B))$
(*iii*) $T(\forall x A)$ iff $(T(A[x \leftarrow t])$ for any closed term t)
(*iv*) $T(\forall X A)$ iff $(T(A[X \leftarrow D])$ for any closed set definition D)

In this definition cases (*i*) to (*iii*) are more or less unavoidable. The main originality of syntactical truth predicates lies in the treatment of the second-order quantification (case (*iv*)): instead of interpreting the quantification over second-order variables by a quantification over sets of natural numbers as in the ordinary semantics of $\boldsymbol{PA^2}$, we quantify over *set definitions*. This is in agreement with the intuitionistic claim that sets do not exist by themselves in a platonistic world but exist only through their definition (for an explanation of this point of view see Poincaré (1913) and Colson and Grigorieff (1998)). Observe that in case (*iv*) the formula $A[X \leftarrow D]$ may be bigger than A and hence makes impossible the construction of such a predicate by induction.

The existence of syntactical truth predicates for implicational formulas has been established in Colson and Grigorieff (1998) using Gödel's notion of constructible set (introduced initially by Gödel to give a proof of consistency of the Axiom of Choice and of the Continuum Hypothesis in Set Theory). It is also shown in a relatively small formal system (*predicative* second-order arithmetic with a comprehension schema limited to arithmetic formulas) that the existence of such a syntactical truth predicate implies the consistency of impredicative second-order arithmetic $\boldsymbol{PA^2}$. This makes the existence theorem 'strong' from a proof-theoretical point of view.

1.2. *Syntactical truth predicates and fixed point theorems*

Syntactical truth predicates for implicational formulas are exactly the sets of closed formula solutions to the fixed point equation $\chi_0(X) = X$, where χ_0 is the function from sets of closed formulas to sets of closed formulas defined by $F \in \chi_0(X)$ iff one of the following conditions is fullfilled:

(*i*) F is $t = u$ and the closed terms t and u have the same value.
(*ii*) F is $A \Rightarrow B$ and $(A \in X) \Rightarrow (B \in X)$.
(*iii*) F is $\forall x A$ and $A[x \leftarrow t] \in X$ for any closed term t.
(*iv*) F is $\forall X A$ and $A[X \leftarrow D] \in X$ for any closed set definition D.

Such a fixed point solution is not immediate to construct: Knaster–Tarski's fixed point theorem (Knaster 1928; Tarski 1955) allows us to build such sets only when the operator (the function χ_0) from sets to sets is *increasing* (or *monotonic* according to the more usual terminology). In the present case χ_0 is not increasing due to the negative occurence of $A \in X$ in condition (*ii*). Elementary attempts to get rid of this problem by a change of language, for instance by switching from the connnective $\Rightarrow$ to the connectives $(\wedge, \neg)$ *fail* (in this last example due to the negative character of the connective $\neg$). To overcome this difficulty, it seems necessary to build a language in which the unavoidable 'negative aspects' have been made as positive as possible. In this paper we propose such a language whose objects are called 'formulas with atomic negation'. These formulas are built with the (positive) connectives $\dot{\wedge}$ and $\dot{\vee}$, and the (positive) first and second-order quantifiers $\dot{\forall}$ and $\dot{\exists}$. No negative connective or quantifier is included in these formulas but instead negation is 'pushed to the atoms', that is, we include atomic formulas $t \dot{=} u$ (for equality) and $t \dot{\neq} u$ (with the positive meaning that t and u are 'separated') and membership/anti-membership atomic formulas $t \dot{\in} X$ and $t \dot{\notin} X$. At this point it is essential to see that any implicational formula can be translated into a formula with atomic negation (such a translation will be detailed in Section 3).

At first sight we have not solved the problem of eliminating the negative aspects of the language since the anti-membership atomic formula $t \dot{\notin} X$ is negative. The crucial remark here is that such membership or anti-membership atomic formulas are *never involved* in the case analysis of the expected definition of a syntactical truth predicate for this language since we deal with *closed formulas* only: in conditions (*i*) to (*iv*) no membership case is considered. This makes the language of formulas with atomic negation especially attractive for constructing a syntactical truth predicate.

1.3. *Contents of this paper*

This paper is organized as follows: we first define formulas with atomic negation. Next we introduce the 'positive negation' $\widetilde{A}$ of such a formula A by induction on A. This allows us to define the second-order substitution in a formula with atomic negation. *Translations* $A \mapsto A^{+}$ from implicational formulas to formulas with atomic negation and $A \mapsto A^{\Rightarrow}$ from formulas with atomic negation to implicational formulas are defined and elementary results concerning them are established. Next, a notion of *syntactical truth predicate for formulas with atomic negation* is defined in accordance with the previous motivation. This allows us to prove in a 'small' formal system (second-order arithmetic with arithmetic comprehension) that there exists a syntactical truth predicate for implicational formulas if and only if there exists a syntactical truth for formulas with atomic negation. Using the results of Colson and Grigorieff (1998), this measures exactly the strength of the existence of such predicates for formulas with atomic negation.

The conclusion discusses the failure of some attempts to construct such objects using the Knaster–Tarski fixed point result, in accordance with the initial motivations.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

2. The language of formulas with atomic negation

2.1. *Formulas with atomic negation*

Working from the motivation given above, we now introduce the notion of a *formula with atomic negation* formally.

Definition 1. The set of *formulas with atomic negation* is defined inductively:

— When t and u are first-order terms, $t \dot{=} u$ is such a formula (with the obvious intended meaning that t and u are equal)
— When t and u are first-order terms, $t \dot{\neq} u$ is such a formula (with the obvious intended meaning that t and u are distinct)
— If t is a first-order term and X a second-order variable, $t \dot{\in} X$ is such a formula (membership statement)
— If t is a first-order term and X a second-order variable, $t \dot{\notin} X$ is such a formula (anti-membership statement)
— When $\boldsymbol{A}$ and $\boldsymbol{B}$ are two such formulas, $\boldsymbol{A} \dot{\wedge} \boldsymbol{B}$ is such a formula (the conjunction of $\boldsymbol{A}$ and $\boldsymbol{B}$)
— When $\boldsymbol{A}$ and $\boldsymbol{B}$ are two such formulas, $\boldsymbol{A} \dot{\vee} \boldsymbol{B}$ is such a formula (the disjunction of $\boldsymbol{A}$ and $\boldsymbol{B}$)
— When $\boldsymbol{A}$ is such a formula, $\dot{\forall} x \boldsymbol{A}$ is such a formula (the first-order universal quantification)
— When $\boldsymbol{A}$ is such a formula, $\dot{\exists} x \boldsymbol{A}$ is such a formula (the first-order existential quantification)
— When $\boldsymbol{A}$ is such a formula, $\dot{\forall} X \boldsymbol{A}$ is such a formula (the second-order universal quantification)
— When $\boldsymbol{A}$ is such a formula, $\dot{\exists} X \boldsymbol{A}$ is such a formula (the second-order existential quantification)

As with the case of implicational formulas, the notion of *set definition with atomic negation* (or simply *set definition*) is defined as an expression $\lambda x \boldsymbol{A}$ where $\boldsymbol{A}$ is a formula with atomic negation. The intended meaning is the same as in the implicational case.

Remarks.

1 In this paper we shall use letters $\boldsymbol{A}$, $\boldsymbol{B}$, ... to stand for such formulas with atomic negation.
2 The usual questions of free and bound variables arise, we will not detail these problems here since they can be treated with this language as in ordinary second-order language.

2.2. *Positive negation*

Definition 2. Let $\boldsymbol{A}$ be a formula with atomic negation. We define the *positive negation* (or more simply the *negation*) $\widetilde{\boldsymbol{A}}$ of $\boldsymbol{A}$ by induction on $\boldsymbol{A}$:

— $\widetilde{t \dot{=} u}$ is $t \dot{\neq} u$
— $\widetilde{t \dot{\neq} u}$ is $t \dot{=} u$

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

— $\widetilde{t\dot{\in}X}$ is $t\dot{\notin}X$
— $\widetilde{t\dot{\notin}X}$ is $t\dot{\in}X$
— $\widetilde{\boldsymbol{A}\dot{\wedge}\boldsymbol{B}}$ is $\widetilde{\boldsymbol{A}}\dot{\vee}\widetilde{\boldsymbol{B}}$
— $\widetilde{\boldsymbol{A}\dot{\vee}\boldsymbol{B}}$ is $\widetilde{\boldsymbol{A}}\dot{\wedge}\widetilde{\boldsymbol{B}}$
— $\widetilde{\dot{\forall}x\boldsymbol{A}}$ is $\dot{\exists}x\widetilde{\boldsymbol{A}}$
— $\widetilde{\dot{\exists}x\boldsymbol{A}}$ is $\dot{\forall}x\widetilde{\boldsymbol{A}}$
— $\widetilde{\dot{\forall}X\boldsymbol{A}}$ is $\dot{\exists}X\widetilde{\boldsymbol{A}}$
— $\widetilde{\dot{\exists}X\boldsymbol{A}}$ is $\dot{\forall}X\widetilde{\boldsymbol{A}}$

We have the following elementary lemma stating that positive negation is involutive.

Lemma 1. Let $\boldsymbol{A}$ be a formula with atomic negation. Then $\widetilde{\widetilde{\boldsymbol{A}}} = \boldsymbol{A}$

Proof. The proof is by induction on $\boldsymbol{A}$. □

2.3. *Substitutions*

The definition of *first-order substitution* $\boldsymbol{A}[x \leftarrow t]$ of a term t for a first-order variable x in a formula with atomic negation $\boldsymbol{A}$ is similar to the definition for implicational formulas: the most interesting case is $(u \in X)[x \leftarrow t]$ defined as $(u[x \leftarrow t]) \in X$. This and the notion of positive negation allows us to give the following definition.

Definition 3. Let $\boldsymbol{A}$ be a formula with atomic negation, X be a second-order variable and $\boldsymbol{D} = \lambda x\boldsymbol{B}$ be a set-definition. We define by induction on $\boldsymbol{A}$ the *substitution* $\boldsymbol{A}[X \leftarrow \boldsymbol{D}]$ of $\boldsymbol{D}$ for X in $\boldsymbol{A}$:

— $(t\dot{=}u)[X \leftarrow \boldsymbol{D}]$ is $t\dot{=}u$.
— $(t\dot{\neq}u)[X \leftarrow \boldsymbol{D}]$ is $t\dot{\neq}u$.
— $(t\dot{\in}Y)[X \leftarrow \boldsymbol{D}]$ is $t\dot{\in}Y$ when Y is different from X.
— The $t\dot{\notin}Y$ case is similar to the previous one.
— $(t\dot{\in}X)[X \leftarrow \boldsymbol{D}]$ is $\boldsymbol{B}[x \leftarrow t]$.
— $(t\dot{\notin}X)[X \leftarrow \boldsymbol{D}]$ is $\widetilde{\boldsymbol{B}}[x \leftarrow t]$.
— $(\boldsymbol{A}\dot{\wedge}\boldsymbol{B})[X \leftarrow \boldsymbol{D}]$ is $(\boldsymbol{A}[X \leftarrow \boldsymbol{D}])\dot{\wedge}(\boldsymbol{B}[X \leftarrow \boldsymbol{D}])$.
— $(\boldsymbol{A}\dot{\vee}\boldsymbol{B})[X \leftarrow \boldsymbol{D}]$ is $(\boldsymbol{A}[X \leftarrow \boldsymbol{D}])\dot{\vee}(\boldsymbol{B}[X \leftarrow \boldsymbol{D}])$.
— $(\dot{\forall}x\boldsymbol{A})[X \leftarrow \boldsymbol{D}]$ is $\dot{\forall}x(\boldsymbol{A}[X \leftarrow \boldsymbol{D}])$.
— The cases of $\dot{\exists}x$, $\dot{\forall}Y$ and $\dot{\exists}Y$ are similar to the previous one.

Remarks.

1 The most interesting point in this definition is the $t\dot{\notin}X$ case since it involves positive negation.

2 In the case of quantifiers, the usual precautions (renaming of bound variables) must be taken to avoid free variable capture during the substitution.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

2.4. *Negation and substitution*

The reader may have noticed that, instead of $\widetilde{\boldsymbol{B}}[x \leftarrow t]$ in the $t \dot{\notin} X$ case of the previous definition, we could have chosen

$$\widetilde{\boldsymbol{B}[x \leftarrow t]}.$$

In fact the two choices are equivalent, as stated by the following lemma.

Lemma 2. Let $\boldsymbol{A}$ be a formula with atomic negation, x be a first-order variable and t be a first-order term. Then

$$(\widetilde{\boldsymbol{A}})[x \leftarrow t] = \widetilde{\boldsymbol{A}[x \leftarrow t]}.$$

Proof. The proof is an elementary induction on $\boldsymbol{A}$. □

We can give an analogous lemma for second-order substitution.

Lemma 3. Let $\boldsymbol{A}$ be a formula with atomic negation, X be a second-order variable and $\boldsymbol{D} = \lambda x \boldsymbol{C}$ be a set definition. Then

$$\widetilde{\boldsymbol{A}[X \leftarrow \boldsymbol{D}]} = (\widetilde{\boldsymbol{A}})[X \leftarrow \boldsymbol{D}].$$

Proof. The proof is by induction on $\boldsymbol{A}$. □

3. From implicational formulas to formulas with atomic negation

In this section we define and study the natural translation from implicational formulas to formulas with atomic negation. The translation is essentially straightforward but needs to be written down precisely in order to prove the substitution lemmas required in sufficient detail.

3.1. *Translation*

Definition 4. Let A be an implicational formula. We define by induction on A the formula with atomic negation A^+ corresponding to A:

— $(t = u)^+$ is $t \dot{=} u$.
— $(t \in X)^+$ is $t \dot{\in} X$.
— $(A \Rightarrow B)^+$ is $\widetilde{(A^+)} \dot{\vee} (B^+)$.
— $(\forall x A)^+$ is $\dot{\forall} x (A^+)$.
— $(\forall X A)^+$ is $\dot{\forall} X (A^+)$.

3.2. *Substitution lemmas*

In this paragraph we state the lemmas relating the previous translation with substitutions. We first have a *first-order substitution lemma.*

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

Lemma 4. Let A be an implicational formula, x be a first-order variable and t be a first-order term. Then we have

$$(A[x \leftarrow t])^+ = (A^+)[x \leftarrow t].$$

Proof. The proof is by induction on A. □

Using the previous lemma we can establish a *second-order substitution lemma.*

Lemma 5. Let A be an implicational formula, X be a second-order variable and $D = \lambda x B$ be a set definition. Then we have

$$(A[X \leftarrow D])^+ = (A^+)[X \leftarrow D^+]$$

where D^+ stands for $\lambda x B^+$.

Proof. The proof is by induction on A. □

3.3. *Translating the negation of a formula*

In the following we borrow from Heyting Arithmetic the following definition of the negation of an implicational formula.

Definition 5. Let A be an implicational formula. We define the (implicational) formula $\neg A$ by the equation $\neg A = (A \Rightarrow (0 = 1))$.

Operating the previous translation on the negation $\neg A$ of some implicational formula A one could expect to obtain $(\neg A)^+ = (\widetilde{A^+})$. However, this is not the case, as shown by the following example: $(\neg(t = u))^+ = ((t = u) \Rightarrow (0 = 1))^+ = (\widetilde{t = u^+})\dot\vee(0\dot=1) = (t\dot\neq u)\dot\vee(0\dot=1)$, which is *not* equal to $(t\dot\neq u)$. However, these two last formulas are 'similar' in the following sense.

Definition 6. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be two formulas with atomic negation. We say that they are *+-similar* and we write $\boldsymbol{A} \sim^+ \boldsymbol{B}$ iff they can be obtained starting from equal formulas by means of substitutions of subformulas $\boldsymbol{C}$ by $\boldsymbol{C}\dot\vee(0\dot=1)$.

Intuitively, such similar formulas have the same meaning.

4. From formulas with atomic negation to implicational formulas

In this section we define and study the converse translation $\boldsymbol{A} \mapsto (\boldsymbol{A}^{\Rightarrow})$ from formulas with atomic negation to implicational formulas.

4.1. *Translation*

In the following definition remember that $\neg A$ stands for $A \Rightarrow (0 = 1)$.

Definition 7. Let $\boldsymbol{A}$ be a formula with atomic negation. We define by induction on $\boldsymbol{A}$ the implicational formula $\boldsymbol{A}^{\Rightarrow}$:

— $(t\dot=u)^{\Rightarrow}$ is $t = u$

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

— $(t \dot{\neq} u)^{\Rightarrow}$ is $\neg(t = u)$
— $(t \dot{\in} X)^{\Rightarrow}$ is $t \in X$
— $(t \dot{\notin} X)^{\Rightarrow}$ is $\neg(t \in X)$
— $(\boldsymbol{A} \dot{\vee} \boldsymbol{B})^{\Rightarrow}$ is $(\neg(\boldsymbol{A}^{\Rightarrow})) \Rightarrow (\boldsymbol{B}^{\Rightarrow})$
— $(\boldsymbol{A} \dot{\wedge} \boldsymbol{B})^{\Rightarrow}$ is $\neg((\boldsymbol{A}^{\Rightarrow}) \Rightarrow \neg(\boldsymbol{B}^{\Rightarrow}))$
— $(\dot{\forall} x \boldsymbol{A})^{\Rightarrow}$ is $\forall x (\boldsymbol{A}^{\Rightarrow})$
— $(\dot{\exists} x \boldsymbol{A})^{\Rightarrow}$ is $\neg \forall x \neg (\boldsymbol{A}^{\Rightarrow})$
— The second-order quantifiers $\dot{\forall} X$ and $\dot{\exists} X$ are translated in the same way as the first-order quantifiers

4.2. *Translating the positive negation of a formula*

In translating the positive negation $\widetilde{\boldsymbol{A}}$ of a formula with positive negation $\boldsymbol{A}$ one could expect to obtain $(\widetilde{\boldsymbol{A}})^{\Rightarrow} = \neg(\boldsymbol{A}^{\Rightarrow})$. However, this is not the case, as is obvious from the following example: $(\widetilde{t \dot{\notin} u})^{\Rightarrow} = (t \dot{\in} u)^{\Rightarrow} = (t = u)$ is different from $\neg((t \dot{\notin} u)^{\Rightarrow}) = \neg(\neg(t \in u))$. However, as in the previous section, these two formulas are similar in the following sense.

Definition 8. Let A and B be two implicational formulas. We say that these formulas are $\Rightarrow$-*similar* and we write $A \sim^{\Rightarrow} B$ if and only if they can be obtained starting with equal formulas by substitutions of some subformulas C by $\neg(\neg C)$.

When no ambiguity is possible we will just write $A \sim B$ and say that A and B are similar. Intuitively again, two $\Rightarrow$-similar formulas have the same meaning. It will turn out that if T is a syntactical truth predicate for implicational formulas and if A and B are two similar formulas, then $T(A)$ is equivalent to $T(B)$ (see Lemma 11).

We can now state the correct lemma concerning negation.

Lemma 6. Let $\boldsymbol{A}$ be a formula with atomic negation. Then $(\widetilde{\boldsymbol{A}})^{\Rightarrow} \sim \neg(\boldsymbol{A}^{\Rightarrow})$.

Proof. The proof is by induction on $\boldsymbol{A}$. □

4.3. *Substitution lemmas*

The following lemmas relate the previous translation to substitution.

Lemma 7. Let $\boldsymbol{A}$ be a formula with atomic negation, x be a first-order variable and t be a first-order term. Then we have

$$(\boldsymbol{A}[x \leftarrow t])^{\Rightarrow} = (\boldsymbol{A}^{\Rightarrow})[x \leftarrow t].$$

Proof. The proof is by induction on $\boldsymbol{A}$. □

Lemma 8. Let $\boldsymbol{A}$ be a formula with atomic negation, X be a second-order variable and $\boldsymbol{D} = \lambda x \boldsymbol{B}$ be a set definition. Then we have

$$(\boldsymbol{A}[X \leftarrow \boldsymbol{D}])^{\Rightarrow} \sim (\boldsymbol{A}^{\Rightarrow})[X \leftarrow \boldsymbol{D}^{\Rightarrow}]$$

where $\boldsymbol{D}^{\Rightarrow}$ stands for $\lambda x(\boldsymbol{B}^{\Rightarrow})$.

Proof. The proof is by induction on $\boldsymbol{A}$. □

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

5. Composing the translations

In this section we shall consider the compositions of the translations $\boldsymbol{A} \mapsto (\boldsymbol{A}^{\Rightarrow})^{+}$ and $A \mapsto (A^{+})^{\Rightarrow}$. We will establish that they are equal to identity up to similarity. Let us first remark that when A is an implicational formula then $(\neg A)^{+}$ is $\widetilde{(A^{+})}\dot{\vee}(0\dot{=}1)$ which is similar to $\widetilde{(A^{+})}$. We shall use this remark in a repeated way.

5.1. *First composition:* $(\boldsymbol{A}^{\Rightarrow})^{+} \sim \boldsymbol{A}$

Lemma 9. Let $\boldsymbol{A}$ be a formula with atomic negation. Then $(\boldsymbol{A}^{\Rightarrow})^{+}$ is $+$-similar to $\boldsymbol{A}$.

Proof. The proof is by induction on $\boldsymbol{A}$. □

5.2. *Second composition:* $(A^{+})^{\Rightarrow} \sim A$

Lemma 10. Let A be an implicational formula. Then $(A^{+})^{\Rightarrow}$ is $\Rightarrow$-similar to A.

Proof. The proof is by induction on A. □

6. Syntactical truth predicates for formulas with atomic negation

As with the case of the implicational language, we can define a notion of syntactical truth predicate for the language of formulas with atomic negation.

Definition 9. Let U be a set of closed formulas with atomic negation. We say that U is a *pre-syntactical truth predicate for formulas with atomic negation* iff the following conditions are fullfilled:

(*i*) $U(t\dot{=}u)$ iff the closed terms t and u have the same value.
(*ii*) $U(t\dot{\neq}u)$ iff the closed terms t and u have different values.
(*iii*) $U(\boldsymbol{A}\dot{\wedge}\boldsymbol{B})$ iff $(U(\boldsymbol{A})$ and $U(\boldsymbol{B}))$.
(*iv*) $U(\boldsymbol{A}\dot{\vee}\boldsymbol{B})$ iff $(U(\boldsymbol{A})$ or $U(\boldsymbol{B}))$.
(*v*) $U(\dot{\forall}x\boldsymbol{A})$ iff $U(\boldsymbol{A}[x \leftarrow t])$ holds for any closed term t.
(*vi*) $U(\dot{\exists}x\boldsymbol{A})$ iff $U(\boldsymbol{A}[x \leftarrow t])$ holds for some closed term t.
(*vii*) $U(\dot{\forall}X\boldsymbol{A})$ iff $U(\boldsymbol{A}[X \leftarrow \boldsymbol{D}])$ holds for any closed set definition $\boldsymbol{D}$.
(*viii*) $U(\dot{\exists}X\boldsymbol{A})$ iff $U(\boldsymbol{A}[X \leftarrow \boldsymbol{D}])$ holds for some closed set definition $\boldsymbol{D}$.

Remarks.

1 As in the case of implicational syntactical truth predicates, this definition is highly circular since (for instance) substituting $\boldsymbol{D}$ for X in $\boldsymbol{A}$ in clause (*vii*) one can meet $\boldsymbol{A}$ again when 'computing' the syntactical truth of $\boldsymbol{A}[X \leftarrow \boldsymbol{D}]$.
2 However, as opposed to the implicational case, we can see that clauses (*i*) to (*viii*) are 'positive' in a sense that we will make precise later. This makes possible the construction by a fixpoint technique of a pre-syntactical truth predicate for formulas with atomic negation, as will be done in Section 7.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

Definition 10. Let U be a set of formulas with atomic negation. We will write $\widetilde{U}$ for the set of formulas with atomic negation defined by

$$(A \in \widetilde{U}) \Longleftrightarrow (\widetilde{A} \in U).$$

Definition 11. Let U be a set of closed formulas with atomic negation.

— We say that U is *coherent* iff for any formula with atomic negation A we have

$$(A \in U) \Rightarrow (\widetilde{A} \notin U).$$

— We say that U is *complete* iff for any formula with atomic negation A we have

$$(A \notin U) \Rightarrow (\widetilde{A} \in U).$$

— We say that U is *symmetrical* iff U is coherent and complete.

Remarks.

1 With the previous notation U coherent means that $U \subset \widetilde{\overline{U}}$ ($\overline{U}$ stands for the set of closed formulas A such that $A \notin U$).

2 U complete means $\widetilde{\overline{U}} \subset U$.

3 U symmetrical means $U = \widetilde{\overline{U}}$.

4 The previous point justifies the terminology 'symmetrical' since U is then invariant under the transformation

$$U \mapsto \widetilde{\overline{U}}.$$

Another possible terminology is 'U *self-dual*'.

We can now give the main definition of this paper.

Definition 12. Let U be a set of closed formulas with atomic negation. We say that U is a *syntactical truth predicate for formulas with atomic negation* iff the following conditions are fullfilled:

(i) U is a pre-syntactical truth predicate for formulas with atomic negation
(ii) U is symmetrical

We will now establish the existence of such a syntactical truth predicate starting with a syntactical truth predicate for implicational formulas and using the previous translations. In fact, we will prove predicatively that the existence of such objects are *equivalent*. Examples of pre-syntactical truth predicates for formulas with atomic negation that are *not* complete or not coherent will be given in Section 7.

6.1. *Existence theorem*

Lemma 11. Let T be a syntactical truth predicate for implicational formulas. Let A and B be two $\Rightarrow$-similar closed implicational formulas. Then $T(A)$ is equivalent to $T(B)$.

Proof. A and B are obtained one from the other by starting from equal formulas and substituting subformulas C by $\neg\neg C$. Hence the formula $A \Longleftrightarrow B$ is provable in classical second-order logic. It follows from the Soundness Lemma of Colson and Grigorieff (1998) that $T(A \iff B)$ holds since T is a syntactical truth predicate. But in this way we can see that $T(A)$ is equivalent to $T(B)$. □

We can now start the construction of a syntactical truth predicate for formulas with atomic negation.

Lemma 12. Let T be a syntactical truth predicate for implicational formulas. Then

$$\boldsymbol{A} \mapsto T(\boldsymbol{A}^{\Rightarrow})$$

is a pre-syntactical truth predicate for formulas with atomic negation.

Proof. We prove that $\boldsymbol{A} \mapsto T(\boldsymbol{A}^{\Rightarrow})$ enjoys conditions (i) to $(viii)$ of Definition 9 by cases on $\boldsymbol{A}$:

— $T((t \dot{=} u)^{\Rightarrow})$ is equivalent to $T(t = u)$: this holds iff t and u have the same value.
— $T((t \dot{\neq} u)^{\Rightarrow})$ is equivalent to $T(\neg(t = u))$, which is equivalent to $\neg T(t = u)$: this holds iff t and u have different values.
— $T((\boldsymbol{A} \dot{\wedge} \boldsymbol{B})^{\Rightarrow})$ is $T(\neg((\boldsymbol{A}^{\Rightarrow}) \Rightarrow \neg(\boldsymbol{B}^{\Rightarrow})))$, which is equivalent to $\neg(T(\boldsymbol{A}^{\Rightarrow}) \Rightarrow \neg T(\boldsymbol{B}^{\Rightarrow}))$, which is equivalent to $T(\boldsymbol{A}^{\Rightarrow}) \wedge T(\boldsymbol{B}^{\Rightarrow})$.
— $T((\boldsymbol{A} \dot{\vee} \boldsymbol{B})^{\Rightarrow})$ is $T(\neg(\boldsymbol{A}^{\Rightarrow}) \Rightarrow (\boldsymbol{B}^{\Rightarrow}))$, which is equivalent to $(\neg T(\boldsymbol{A}^{\Rightarrow})) \Rightarrow T(\boldsymbol{B}^{\Rightarrow})$, which is equivalent to $T(\boldsymbol{A}^{\Rightarrow}) \vee T(\boldsymbol{B}^{\Rightarrow})$.
— $T((\dot{\forall} x \boldsymbol{A})^{\Rightarrow})$ is $T(\dot{\forall} x(\boldsymbol{A}^{\Rightarrow}))$, which is equivalent to $\forall t$ closed $T(\boldsymbol{A}^{\Rightarrow}[x \leftarrow t])$, which is equivalent to $\forall t$ closed $T((\boldsymbol{A}[x \leftarrow t])^{\Rightarrow})$ by Lemma 7.
— $T((\dot{\exists} x \boldsymbol{A})^{\Rightarrow})$ is $T(\neg(\forall x \neg(\boldsymbol{A}^{\Rightarrow})))$, which is equivalent to $\neg \forall t$ closed $\neg T(\boldsymbol{A}^{\Rightarrow}[x \leftarrow t])$, which is equivalent to $\exists t$ closed $T(\boldsymbol{A}^{\Rightarrow}[x \leftarrow t])$, which is equivalent to $\exists t$ closed $T((\boldsymbol{A}[x \leftarrow t])^{\Rightarrow})$.
— $T((\dot{\forall} X \boldsymbol{A})^{\Rightarrow})$ is $T(\forall X(\boldsymbol{A}^{\Rightarrow}))$, which is equivalent to

$$\forall D \text{ closed } T(\boldsymbol{A}^{\Rightarrow}[X \leftarrow D]). \tag{1}$$

We have to prove that this is equivalent to $T(\boldsymbol{A}[X \leftarrow \boldsymbol{E}]^{\Rightarrow})$ for any closed set definition $\boldsymbol{E}$. But if (1) holds and if $\boldsymbol{E}$ is such a definition, then $T(\boldsymbol{A}^{\Rightarrow}[X \leftarrow \boldsymbol{E}^{\Rightarrow}])$ holds. Hence $T(\boldsymbol{A}[X \leftarrow \boldsymbol{E}]^{\Rightarrow})$ by the substitution lemma (Lemma 8) and the previous lemma. Conversely, if $T(\boldsymbol{A}[X \leftarrow \boldsymbol{E}]^{\Rightarrow})$ holds for any closed set definition $\boldsymbol{E}$, let D be a set definition like in (1). We then have $T(\boldsymbol{A}[X \leftarrow D^{+}]^{\Rightarrow})$, and hence by the substitution lemma $T(\boldsymbol{A}^{\Rightarrow}[X \leftarrow D^{+^{\Rightarrow}}]$. But, by the second composition lemma (Lemma 10), D and $D^{+^{\Rightarrow}}$ are similar, hence $\boldsymbol{A}^{\Rightarrow}[X \leftarrow D]$ and $\boldsymbol{A}^{\Rightarrow}[X \leftarrow D^{+^{\Rightarrow}}]$ are also similar. Thus, by the previous lemma, we get $T(\boldsymbol{A}^{\Rightarrow}[X \leftarrow D])$
— The second-order existential quantifier case is essentially similar to the previous one ($\forall D$ and $\forall \boldsymbol{E}$ are replaced by $\exists D$ and $\exists \boldsymbol{E}$). □

Hence the first condition to be a syntactical truth predicate for formulas with atomic negation is fullfilled. Let us now examine the second condition.

Lemma 13. Let T be a syntactical truth predicate for implicational formulas. Then $\boldsymbol{A} \mapsto (\boldsymbol{A}^{\Rightarrow})$ is symmetrical.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

Proof. Let $\boldsymbol{A}$ be a formula with atomic negation. We have to prove that $T((\widetilde{\boldsymbol{A}})^{\Rightarrow})$ holds if and only if $\neg T(\boldsymbol{A}^{\Rightarrow})$ holds. But, by Lemma 6, $(\widetilde{\boldsymbol{A}})^{\Rightarrow}$ is similar to $\neg(\boldsymbol{A}^{\Rightarrow})$, hence the result follows by the definition of syntactical truth predicates for implicational formulas and by Lemma 11. □

We can now conclude our first existence result (remember that existence of syntactical truth predicates for implicational formulas was established in Colson and Grigorieff (1998)).

Proposition 1. Let T be a syntactical truth predicate for implicational formulas. Then $\boldsymbol{A} \mapsto (\boldsymbol{A}^{\Rightarrow})$ is a syntactical truth predicate for formulas with atomic negation.

Proof. The proof follows from the last two lemmas. □

Corollary 1. The existence of a syntactical truth predicate for formulas with atomic negation follows from the existence of a syntactical truth predicate for implicational formulas.

Remark. Notice that this last result was established by predicative means: most lemmas are formalizable in first-order Peano arithmetic via a primitive recursive encoding of terms and formulas by natural numbers. Only the definitions and lemmas of this section involved second-order objects (sets of formulas, that is, sets of natural numbers). It is essential at this point to recognize that *arithmetic comprehension* was enough for our purpose: that is, for instance, passing from $A[X \leftarrow D]$ to $\exists XA$ in the last corollary in the set definition D was *without* any second-order quantifiers.

6.2. *From predicates for formulas with atomic negation to implicational syntactical truth predicates*

We now establish the converse result: from the existence of a syntactical truth predicate for formulas with atomic negation we infer the existence of a syntactical truth predicate for implicational formulas. This makes the existence of a syntactical truth predicate for formulas with atomic negation 'strong' from a proof-theoretical point of view since such a result easily entails the consistency of second-order arithmetic.

We start with a result similar to Lemma 11.

Lemma 14. Let U be a syntactical truth predicate for formulas with atomic negation. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be two $+$-similar closed formulas with atomic negation. Then $U(\boldsymbol{A})$ is equivalent to $U(\boldsymbol{B})$.

Proof. We will not develop either a proof system for the language of formulas with atomic negation or a soundness lemma analogous to the one we used in the proof of Lemma 11. We will do the proof 'by hand' instead. $\boldsymbol{A}$ and $\boldsymbol{B}$ are similar formulas, hence they are obtained one from the other by means of substitutions of subformulas $\boldsymbol{C}$ by $\boldsymbol{C}\dot{\vee}(0\doteq1)$. Without loss of generality, we can assume that only one such substitution has been performed. So, let us assume that $\boldsymbol{B}$ has been obtained from $\boldsymbol{A}$ by substituting a

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

subformula $\boldsymbol{C}$ by $\boldsymbol{C}\dot{\vee}(0\dot{=}1)$. Here we shall use some intuition about syntactical truth predicates. The 'computation' of the syntactical truth $U(\boldsymbol{A})$ and $U(\boldsymbol{B})$ will start with the same steps until we meet $\boldsymbol{C}$ and $\boldsymbol{C}\dot{\vee}(0\dot{=}1)$. At this point some substitutions of x by a closed term t or of X by a closed definition $\boldsymbol{D}$ in $\boldsymbol{C}$ may have been performed when going through a quantifier $\dot{\forall}x$, $\dot{\exists}x$ or $\dot{\forall}X$, $\dot{\exists}X$. Hence $\boldsymbol{C}$ may have been transformed into $\boldsymbol{C}'$ and we have to 'compute' the syntactical truth of $\boldsymbol{C}'$ and $\boldsymbol{C}'\dot{\vee}(0\dot{=}1)$. But U is a syntactical truth predicate, thus we have $U(\boldsymbol{C}'\dot{\vee}(0\dot{=}1)) \Leftrightarrow U(\boldsymbol{C}')$, and hence $U(\boldsymbol{A}) \Leftrightarrow U(\boldsymbol{B})$. □

The precise formalization of this proof is as follows.

Lemma 15. Let U be a syntactical truth predicate for formulas with atomic negation. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be two $+$-similar formulas with atomic negation with variables among $\vec{x}$ and $\vec{X}$. Let $\vec{t}$ and $\vec{\boldsymbol{D}}$ be sequences of closed terms and set definitions adapted for $\vec{x}$ and $\vec{X}$. Then $U(\boldsymbol{A}[\vec{x} \leftarrow \vec{t}, \vec{X} \leftarrow \vec{\boldsymbol{D}}])$ is equivalent to $U(\boldsymbol{B}[\vec{x} \leftarrow \vec{t}, \vec{X} \leftarrow \vec{\boldsymbol{D}}])$.

Proof. The proof is by a simultaneous induction on $\boldsymbol{A}$ and $\boldsymbol{B}$. □

We can now construct a syntactical truth predicate for implicational formulas from a syntactical truth predicate for formulas with atomic negation.

Proposition 2. Let U be a syntactical truth predicate for formulas with atomic negation. Then $A \mapsto U(A^+)$ is a syntactical truth predicate for implicational formulas.

Proof. The proof is by cases on A:

- $U((t = u)^+)$ is $U(t\dot{=}u)$, which holds iff t and u have the same value
- $U((A \Rightarrow B)^+)$ is $U(\widetilde{A}\dot{\vee}B)$, which is equivalent to $U(\widetilde{A}) \vee U(B)$ (since U is a pre-syntactical truth predicate), which is equivalent to $\neg U(A) \vee U(B)$ (since U is symmetrical), which is equivalent to $U(A) \Rightarrow U(B)$
- $U((\forall x A)^+)$ is $U(\dot{\forall}x(A^+))$, which is equivalent to $\forall t$ closed $U(A^+[x \leftarrow t])$, which is equivalent to $U((A[x \leftarrow t])^+)$ by Lemma 4
- $U((\forall X A)^+)$ is $U(\dot{\forall}X(A^+))$, which is equivalent to

$$\forall \boldsymbol{D} \text{ closed } U(A^+[X \leftarrow \boldsymbol{D}]) \tag{1}$$

  We have to prove that this is equivalent to $\forall E\ U((A[X \leftarrow E])^+)$. But if E is such a definition, then, taking E^+ for $\boldsymbol{D}$, one gets by (1) that $U(A^+[X \leftarrow E^+])$, that is, $U((A[X \leftarrow E])^+)$ by Lemma 5. Conversely, if $\forall E\ U((A[X \leftarrow E])^+)$ holds and if $\boldsymbol{D}$ is a definition like in (1), we have $U((A[X \leftarrow \boldsymbol{D}^{\Rightarrow}])^+)$, which is equivalent to $U(A^+[X \leftarrow \boldsymbol{D}^{\Rightarrow +}])$. But $\boldsymbol{D}^{\Rightarrow +}$ is similar to $\boldsymbol{D}$ by Lemma 9, and hence, by the previous lemma, we get $U(A^+[X \leftarrow \boldsymbol{D}])$ □

7. Syntactical truth predicates as fixpoints

In this section we discuss some naive attempts to build syntactical truth predicates for formulas with atomic negation by means of the Knaster–Tarski fixed point theorem, and why they fail.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

7.1. *An increasing operator on sets of closed formulas*

Definition 13. Let X be a set of closed formulas with atomic negation. We define the set of closed fomulas with atomic negation $\varphi_0(X)$ by $\boldsymbol{A} \in \varphi_0(X)$ if and only if one of the following conditions is fullfilled:

(i) $\boldsymbol{A}$ is $t \dot{=} u$ and the closed terms t and u have the same value.
(ii) $\boldsymbol{A}$ is $t \dot{\neq} u$ and the closed terms t and u have different values.
(iii) $\boldsymbol{A}$ is $\boldsymbol{B} \dot{\wedge} \boldsymbol{C}$ and ($X(\boldsymbol{A})$ and $X(\boldsymbol{B})$).
(iv) $\boldsymbol{A}$ is $\boldsymbol{B} \dot{\vee} \boldsymbol{C}$ and ($X(\boldsymbol{A})$ or $X(\boldsymbol{B})$).
(v) $\boldsymbol{A}$ is $\dot{\forall} x \boldsymbol{B}$ and $X(\boldsymbol{B}[x \leftarrow t])$ holds for any closed term t.
(vi) $\boldsymbol{A}$ is $\dot{\exists} x \boldsymbol{B}$ and $X(\boldsymbol{B}[x \leftarrow t])$ holds for some closed term t.
(vii) $\boldsymbol{A}$ is $\dot{\forall} Y \boldsymbol{B}$ and $X(\boldsymbol{A}[Y \leftarrow \boldsymbol{D}])$ holds for any closed set definition $\boldsymbol{D}$.
($viii$) $\boldsymbol{A}$ is $\dot{\exists} Y \boldsymbol{A}$ and $X(\boldsymbol{A}[Y \leftarrow \boldsymbol{D}])$ holds for some closed set definition $\boldsymbol{D}$.

Lemma 16. Let U be a set of closed formulas with atomic negation. Then U is a pre-syntactical truth predicate if and only if U is a fixed point of φ_0 (that is, $\varphi_0(U) = U$).

Proof. The proof is immediate from the definitions. □

Lemma 17. The operator φ_0 is increasing, that is, for any sets X, Y of closed formulas with atomic negation $X \subseteq Y$ implies $\varphi_0(X) \subseteq \varphi_0(Y)$.

Proof. The proof is immediate from the shape of the definition of φ_0: all occurences of X in this definition are positive. □

As a corollary, observe that one can build in such a situation a pre-syntactical truth predicate by means of the Knaster–Tarski fixed point theorem. We shall now focus on two such solutions: the least and the greatest fixed point of φ_0.

7.2. *Two fixed point solutions*

Definition 14. Let δ be an operator (that is, a function from sets of closed formulas to sets of closed formulas). We define the sets $\mathit{fix}_X \delta(X)$ and $\mathit{Fix}_Y \delta(Y)$ by the equations:

— $\mathit{fix}_X \delta(X) = \bigcap \{X | \delta(X) \subseteq X\}$.
— $\mathit{Fix}_Y \delta(Y) = \bigcup \{Y | Y \subseteq \delta(Y)\}$.

We recall the following classical results.

Proposition 3. (Tarski) Let δ be an increasing operator. Then $\mathit{fix}_X \delta(X)$ and $\mathit{Fix}_Y \delta(Y)$ are fixed points of δ. Furthermore, $\mathit{fix}_X \delta(X)$ is the least fixed point of δ and $\mathit{Fix}_Y \delta(Y)$ is the greatest fixed point of δ.

Proof. Consider the complete lattice of sets of closed formulas (ordered by inclusion) and apply the results of Tarski (1955). □

Definition 15. In the following X_0 will stand for $\mathit{fix}_X \varphi_0(X)$ and Y_0 for $\mathit{Fix}_Y \varphi_0(Y)$.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

Using the previous results, it is clear that X_0 and Y_0 are pre-syntactical truth predicates. In order to build a syntactical truth predicate, a symmetry condition is required, which we shall examine now.

The following lemma is part of the folklore of fixed point theory.

Lemma 18. Let δ be an increasing operator on sets of closed formulas with atomic negation. Then

$$\overline{\mathit{fix}_X \delta(X)} = \mathit{Fix}_X \overline{\delta(\overline{X})}$$

The following is another easy lemma.

Lemma 19. Let δ be an increasing operator on sets of closed formulas with atomic negation. Let X_δ be the set $\mathit{fix}_X \delta(X)$. Then

$$\widetilde{X_\delta} = \mathit{fix}_X \widetilde{\delta(\widetilde{X})}$$

Proof. We have $X_\delta = \bigcap\{X | \delta(X) \subseteq X\}$. By definition of $X \mapsto \widetilde{X}$, it is easy to see that $\widetilde{X_\delta} = \bigcap\{\widetilde{X} | \delta(X) \subseteq X\}$. But the transformation $X \mapsto \widetilde{X}$ is involutive (that is, $\widetilde{\widetilde{X}} = X$), hence we have, by putting $X' = \widetilde{X}$,

$$\widetilde{X_\delta} = \bigcap\{X' | \delta(\widetilde{X'}) \subseteq \widetilde{X'}\}.$$

But clearly $\delta(\widetilde{X'}) \subseteq \widetilde{X'}$ is equivalent to $\widetilde{\delta(\widetilde{X'})} \subseteq X'$, which gives the result. □

Similarly, we can prove the following lemma.

Lemma 20. Let δ be an increasing operator. Let Y_δ be the set $\mathit{Fix}_Y \delta(Y)$. Then

$$\widetilde{Y_\delta} = \mathit{Fix}_Y \widetilde{\delta(\widetilde{Y})}$$

Plugging the last two results together we get the following lemma.

Lemma 21. $\widetilde{\overline{X_0}} = \mathit{Fix}_X \overline{\left(\widetilde{\varphi_0(\widetilde{\overline{X}})}\right)}$ and $\widetilde{\overline{Y_0}} = \mathit{fix}_Y \overline{\left(\widetilde{\varphi_0(\widetilde{\overline{Y}})}\right)}$

Proof. The proof follows immediately from the last two lemmas. □

Definition 16. Let φ be an operator on sets of closed formulas with atomic negation. We say that φ is *1-symmetrical* iff $\varphi(\widetilde{\overline{X}}) = \widetilde{\overline{\varphi(X)}}$.

Lemma 22. The operator φ_0 is 1-symmetrical.

Proof. The proof is by a case analysis on the formulas. For instance, for the conjunctive case, $(\boldsymbol{A} \dot{\wedge} \boldsymbol{B}) \in \varphi_0(\widetilde{\overline{X}})$ is equivalent to $(\boldsymbol{A} \in \widetilde{\overline{X}}$ and $\boldsymbol{B} \in \widetilde{\overline{X}})$, which is equivalent to $(\widetilde{\boldsymbol{A}} \notin X$ and $\widetilde{\boldsymbol{B}} \notin X)$, which is equivalent to $\neg(\widetilde{\boldsymbol{A}} \in X$ or $\widetilde{\boldsymbol{B}} \in X)$, which is equivalent to $\neg((\widetilde{\boldsymbol{A}} \dot{\vee} \widetilde{\boldsymbol{B}}) \in \varphi_0(X))$, which is equivalent to $(\widetilde{\boldsymbol{A} \dot{\wedge} \boldsymbol{B}}) \notin \varphi_0(X)$, which is equivalent to $(\boldsymbol{A} \dot{\wedge} \boldsymbol{B}) \in \widetilde{\overline{\varphi_0(X)}}$ □

Corollary 2. The sets X_0 and Y_0 enjoy the equation $\widetilde{\overline{X_0}} = Y_0$.

Proof. The proof follows from the last two lemmas. □

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

Corollary 3. X_0 is coherent and Y_0 is complete.

Proof. The least fixed point of an operator is always included in the greatest fixed point of this operator, hence we have $X_0 = \mathit{fix}_X\,\varphi_0(X) \subseteq \mathit{Fix}_Y\,\varphi_0(Y) = Y_0 = \widetilde{X_0}$ (this shows that X_0 is coherent), and, similarly, $\widetilde{Y_0} = X_0 \subseteq \widetilde{X_0} = Y_0$ (hence Y_0 is complete). □

We shall now show, however, that X_0 and Y_0 are *not* syntactical truth predicates for formulas with atomic negation.

Definition 17. Let $\perp\!\!\!\perp$ and $\top\!\!\!\top$ be the formulas defined by:
— $\perp\!\!\!\perp = \dot\exists X \dot\exists x (x \dot\in X) \dot\wedge (x \dot\notin X)$.
— $\top\!\!\!\top = \dot\forall X \dot\forall x (x \dot\in X) \dot\vee (x \dot\notin X)$.

Remark. $\perp\!\!\!\perp$ and $\top\!\!\!\top$ are a definition of falsity and truth by means of *pure second-order* formulas with atomic negation, that is, without an equality statement.

Lemma 23. Let U be a syntactical truth predicate for formulas with atomic negation. Then $U(\top\!\!\!\top)$ holds and $U(\perp\!\!\!\perp)$ does not hold.

Proof. We show, for instance, that $U(\top\!\!\!\top)$ holds (the other proof can be obtained by duality). $U(\top\!\!\!\top)$ is

$$U(\dot\forall X \dot\forall x (x \dot\in X) \dot\vee (x \dot\notin X)),$$

which is equivalent to

$$\forall \boldsymbol{D}\, U(\dot\forall x \boldsymbol{D}(x) \dot\vee \widetilde{\boldsymbol{D}}(x))$$

(where $\boldsymbol{D}(x)$ stands for $\boldsymbol{A}[y \leftarrow x]$ when $\boldsymbol{D}$ is $\lambda y \boldsymbol{A}$), which is equivalent to

$$\forall \boldsymbol{D} \forall t\, U(\boldsymbol{D}(t) \dot\vee \widetilde{\boldsymbol{D}}(t)),$$

which is equivalent to

$$\forall \boldsymbol{D} \forall t\, U(\boldsymbol{D}(t)) \vee U(\widetilde{\boldsymbol{D}(t)}),$$

which holds since U is symmetrical. □

However, we have the following for X_0 and Y_0.

Lemma 24. $X_0(\top\!\!\!\top)$ does not hold, but $Y_0(\perp\!\!\!\perp)$ holds.

Proof. Let $X_0' = X_0 - \{\top\!\!\!\top, \dot\forall x(\top\!\!\!\top \dot\vee \perp\!\!\!\perp), \top\!\!\!\top \dot\vee \perp\!\!\!\perp\}$. We claim that $\varphi_0(X_0') \subseteq X_0'$ (this will establish that $X_0 \subseteq X_0'$ by definition of X_0 and hence that $\top\!\!\!\top \notin X_0$). We already know that $X_0' \subseteq X_0$, and hence that $\varphi_0(X_0') \subseteq \varphi_0(X_0) \subseteq X_0$. We thus have to check that $\top\!\!\!\top, \dot\forall x(\top\!\!\!\top \dot\vee \perp\!\!\!\perp), \top\!\!\!\top \dot\vee \perp\!\!\!\perp$ do not belong to $\varphi_0(X_0')$.
— Assume that $\top\!\!\!\top \in \varphi_0(X_0')$, that is,

$$(\dot\forall X \dot\forall x (x \dot\in X) \dot\vee (x \dot\notin X)) \in \varphi_0(X_0').$$

By definition of φ_0 we must have for a particular closed set definition $\boldsymbol{D}$ that

$$(\dot\forall x \boldsymbol{D}(x) \dot\vee \widetilde{\boldsymbol{D}(x)}) \in X_0'.$$

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press

Taking $\boldsymbol{D} = \lambda x \top$, we see that we must have

$$(\dot{\forall} x(\top \dot{\vee} \perp)) \in X_0',$$

which is impossible by definition of X_0'.

— Assume that

$$(\dot{\forall} x(\top \dot{\vee} \perp)) \in \varphi_0(X_0').$$

We must then have

$$(\top \dot{\vee} \perp) \in X_0',$$

which is impossible by definition of X_0'.

— Assume that

$$(\top \dot{\vee} \perp)) \in \varphi_0(X_0').$$

We must then have $\top \in X_0'$ or $\perp \in X_0'$. We already know by the previous corollary that X_0 is coherent and that X_0 is a fixed point of φ_0. It is then easy to see that $\perp$ cannot belong to X_0 and hence to X_0'. In this way we get that $\top \in X_0'$, which is impossible by definition of X_0'. □

Conclusion

We have defined the language of formulas with atomic negation, which eliminates as far as possible the negative aspects of the ordinary second-order language. We have next defined syntactical truth predicates for this language as fixed points of an increasing operator having some symmetry property. The existence of such truth predicates has the same strength as the corresponding notion for the more ordinary language of implicational formulas.

We have shown how the most elementary attempts to build such a syntactical truth predicate by a fixed point technique fail. These attempts are, however, quite natural and suggest that this approach could succeed using more sophisticated fixed points theorems needing some higher-order arithmetic. Our experience of this problem is that one difficulty in this direction is the weakly mathematical aspect of the notion of formula: proofs and definitions concerning them are usually tedious inductions or case analyses. A more mathematical notion of formula than the one considered in this paper could then perhaps lead to the construction of a symmetrical fixed point.

Acknowledgements

I would like to thank Thierry Coquand, Serge Grigorieff and Yuri Gurevich for discussions concerning the notion of formulas with atomic negation and their syntactical truth predicates.

References

Takeuti, G. (1975) *Proof theory*, North-Holland.

Colson, L. and Grigorieff, S. (1998) Syntactical truth predicates for second-order arithmetic. To appear in the *Journal of Symbolic Logic*.

Poincaré, H. (1913) *Dernières pensées*, Flammarion, Paris. (English translation in Last thoughts, Dover Publications, 1963.)

Knaster, B. (1928) Un théorème sur les fonctions d'ensembles. *Ann. Soc. Pol. Math.* **6** 133–134.

Tarski, A. (1955) A lattice-theoretical fixed point theorem and its applications. *Pacif. J. Math.* **5** 285–309.

https://doi.org/10.1017/S0960129501003449 Published online by Cambridge University Press