AN INTRODUCTION TO THE TECHNIQUE OF A FOUNDATION OF ELEMENTARY ARITHMETIC

by Andrew Boucher

A Foundation of Elementary Arithmetic proves, in rigorous fashion, the theorems of elementary arithmetic. This summary is meant to explain how to follow the proofs, and should therefore be read first. An introduction to the philosophical underpinnings of the system can be found in “A Philosophical Introduction to A Foundation for Elementary Arithmetic” and will deepen one’s understanding, although its reading is not strictly a prerequisite.

A Natural-Deduction Logic

A Foundation for Elementary Arithmetic uses a natural-deduction logic, widely disseminated by Lemmon’s standard text, *Beginning Logic*. For example, if P and Q have been proved, then P&Q can be asserted, justified by the rule &I, i.e. & Introduction.

Rules vs Style

The FOEA system borrows several concepts from computer programming. One of the most basic is that of the language rules, which are rigidly defined, versus the style, which is left to the individual but which is expected to be as perspicacious as possible. For instance, in the computing language C, it is a rule that

```
if (A) B
```

is a statement if A is an expression and B is a statement. It is, however, a stylistic preference whether one writes in one’s program

```
if (a <= b) c = b;
```

or

```
if (a <= b)
  c = b;
```

or

```
if (a <= b)
```

```
        c = b;
```

Any of these three variations are correct. But a programmer, once having chosen a style, is expected to use it consistently throughout the program, to render it more readable.

According to the rules of FOEA logic, whitespace is not important, and so both

$$\forall n\forall m(\leq[n,m]\Rightarrow\omega[n]\&\omega[m])$$

and

$$\forall n\forall m \ (\ \leq[n,m] \ \Rightarrow \ \omega[n] \ \& \ \omega[m] \ )$$

are equivalent statements. However, it is a stylistic device that whitespace will be used plentifully and for the most part consistently, and so usually only the second will actually be found in FOEA.

Another example: It is a rule in the FOEA logic that both normal and bold letters can be used as variables. It is however a stylistic device, uniformly applied throughout FOEA, that only non-bold letters are bound variables.

Comments

Another notion of computer science which has been borrowed and used extensively is that of the comment. Comments are not part of the proofs, and theoretically could be eliminated. Their purpose is to provide the reader with an additional means to understand the proceedings or to provide a deeper comprehension. Evidently, there can always be more comments, but some care has been taken to well document each chapter.

Comments must begin with an explanation point (!) and end with an inverted explanation point (¡).

A Simple Proof

Without further ado, herein is an actual proof of FOEA, from Section V, Chapter 2 ("ω[n]" means that "n is a natural number", "¬"

means “not”, and “m'” refers to the successor of m, i.e. m+1):

! **49.** P49 is yet another way of saying that every non-zero finite number has a finite predecessor. ¡

⊦ ∀n (ω[n] & ¬ n = 0 ⇒ ∃m (m + 1) = n) ¡

n	,! 1 (Prem)	¡
ω[**n**] & ¬ **n** = 0	,! 2 (Prem)	¡
(ω[**n**] & ¬ **n** = 0 ⇒ ∃m (m') = **n**)	,! 3 (∀E: IV8.34)	¡
ω[**n**] & ¬ **n** = 0 ⇒ ∃m (m') = **n**	,! 4 (()E: 3)	¡
∃m (m') = **n**	,! 5 (⇒E: 2,4)	¡
(**m**') = **n**	,! 6 (∃E: 5)	¡
((**m**') = **n** ⇒ (**m** + 1) = **n**)	,! 7 (∀E: P44)	¡
(**m**') = **n** ⇒ (**m** + 1) = **n**	,! 8 (()E: 7)	¡
(**m** + 1) = **n**	,! 9 (⇒E: 6,8)	¡
∃m (m + 1) = **n**	,! 10 (∃I: 9)	¡
ω[**n**] & ¬ **n** = 0 ⇒ ∃m (m + 1) = **n**	,! 11 (⇒I: 2,10)	¡
(ω[**n**] & ¬ **n** = 0 ⇒ ∃m (m + 1) = **n**)	,! 12 (()I: 11)	¡
∀n (ω[n] & ¬ n = 0 ⇒ ∃m (m + 1) = n)	! 13 (∀I: 1,12)	¡

▯

The first line,

! **49.** P49 is yet another way of saying that every non-zero finite number has a finite predecessor. ¡

is a comment, which gives the proposition’s number (49) and explains what the proposition is assertion. Every proposition in FOEA will have a number, so it can be referenced easily.

The next line

$\vdash \forall n\ (\ \omega[n]\ \&\ \neg\ n = 0 \Rightarrow \exists m\ (m + 1) = n\ )$ ¡

beginning with a turnstile "⊦" and finishing with an inverted explanation point "¡", is the *header*, which states the proposition to be proved, namely that every non-zero natural number is equal to something plus one. Note the stylistic device of whitespace between the quantifier block and the first parenthesis, the first parenthesis and "ω", and "n" and the last parenthesis. Also note that the quantified "n" and "m" are not bold.

The first line of the proof is:

n ,! 1 (Prem) ¡

It says, "Suppose **n** exists," and, for computer programmers, may be thought of as corresponding to a local declaration.

In fact, the only symbols from this line which are part of the proof are "**n**" and ",". And in fact the most parsimonious transcription would have been:

n,

The actual manner of writing it takes into account several standard stylistic devices. First, the comma, which separates the various statements of a proof (but is not itself part of a statement), always appears throughout FOEA in the same column on the right-hand side. So there is whitespace between the **n** and the comma. Secondly, each line is always given a comment which indicates the line's number and the justification for the line, in this case Prem. Thirdly, this comment block always begins in the same column. Fourthly, a line justified by Prem is always indented 2 spaces more than the previous line or, if there is no previous line, it begins in column 3. And fifthly, **n** is written in bold. Technically the bold and non-bold versions of a letter have no connection. According to the rules of the language, "**m**" or indeed "m" (both acceptable variables) could have been written just as well as "**n**". However, generally the bold version of a letter will be used when it corresponds to a particular universally quantified variable.

Line 2

ω[**n**] & ¬ **n** = 0 ,! 2 (Prem) ¡

is indented 2 more spaces than the preceding line (and so begins in column 5). It says, “Suppose that **n** is a natural number and that **n** does not equal 0.”

Remark that “Prem” is thus overloaded and has two distinct, although similar uses, as shown by lines 1 and 2. Its use in the first line may be called “Prem Type I,” that in the second “Prem Type II.”

Line 3

(ω[**n**] & ¬ **n** = 0 ⇒ ∃m (m') = **n**) ,! 3 (∀E: IV8.34) ¡

is justified by the ∀E (elimination) rule, using proposition number 34

∀n (ω[n] & ¬ n = 0 ⇒ ∃m (m') = n)

which was already proved in Section IV, Chapter 8. Remark that if “∀E: C1.34” had appeared, i.e. no roman numerals, the section is that in which the proof appears; since the proof is Section V, the proposition referred to would have appeared in Section V, Chapter 1.

Line 4 eliminates the parentheses of line 3.

Line 5 follows from lines 2 and 4, using *modus ponens* or, as it is called in a natural-deduction system, “⇒E” (Elimination).

The sixth line

(**m**') = **n** ,! 6 (∃E: 5) ¡

follows from the fifth. If there exists an m such that (m') = n, then the rule “∃E” allows us to introduce a new letter, always in bold-case by style, replacing the “m”. Again, we could have used a different letter than a bold “**m**”; the choice was stylistic.

Line 7

$((\mathbf{m}') = \mathbf{n} \Rightarrow (\mathbf{m} + 1) = \mathbf{n})$,! 7 ($\forall$E: P44) ¡

is justified by the “$\forall$E” rule and a previously proven proposition,

$\forall n \forall m\ ((m') = n \Rightarrow (m + 1) = n)$

The “P44” means that it can be found in the same section and chapter, i.e. it is proposition number 44 of Section V, Chapter 2.

Parentheses are removed and *modus ponens* applied in lines 8 and 9. Line 10 applies the “$\exists$I” rule to line 9.

The eleventh line

$\omega[\mathbf{n}]\ \&\ \neg\ \mathbf{n} = 0 \Rightarrow \exists m\ (m + 1) = \mathbf{n}$,! 11 ($\Rightarrow$I: 2,10) ¡

is indented 2 less than the previous line, which is the standard style for the “$\Rightarrow$I” rule. The “$\Rightarrow$I” can be used only when there has been a previous statement introduced using the “Prem Type II” rule, line 2 in this case, and this premised statement must appear preceding the “$\Rightarrow$“. Succeeding the “$\Rightarrow$“ must be the statement of the previous line, in this case line 10.

Remark that all lines from 2 to 10, which all have the same indentation, are in effect all part of the same hypothetical world: where **n** exists and where $\omega[\mathbf{n}]\ \&\ \neg\ \mathbf{n} = 0$. Line 11 is the first line no longer in this world or environment, supposing no longer that $\omega[\mathbf{n}]\ \&\ \neg\ \mathbf{n} = 0$ (but still supposing that **n** exists).

Line 12 adds back parentheses.

Line 13

$\forall n\ (\ \omega[n]\ \&\ \neg\ n = 0 \Rightarrow \exists m\ (m + 1) = n\ )$! 13 ($\forall$I: 1,12) ¡

does not have a comma, being the last statement of the proof and thus not needing separation from a succeeding statement. It is indented two less than the previous, which is standard for statements justified by "∀I". Indeed, this rule applies only when there have been symbols introduced by the "Prem Type I" rule, as **n** was in line 1.

Remark that this line is the only one beginning in column 1. The last line in a proof cannot be indented, and while not the case for the proof duplicated here, it happens (not often, but sometimes) that other lines in a proof may also have 0 indentation.

The solitary symbol "□" at the end is used to indicate that the proof is terminated.

A Dynamic Language

The language of FOEA is dynamic, in the sense the language at the end of the work is richer--has more symbols--that that at the beginning. The number of potential symbols may be supposed either finite or not, depending on the predilection of the reader.

New symbols are added to the language using one of three rules, 𝕊, 𝔻, or 𝕋. The 𝕊 and 𝔻 rules allow for abbreviations, the first for statements, the second for terms. The 𝕋 rule applies to terms.

The 𝕊 Definition

An example of an 𝕊 definition is:

! **1.** ⊆ represents inclusion (of one-place predicates). ¡

𝕊 ⊆ ; **P** ⊆ **Q** ; ∀x(**P**[x] ⇒ **Q**[x]) ¡

Remark a comment precedes the definition, with its obligatory number.

In the definition proper, the new symbol, in this case "⊆", follows directly after "𝕊" and is followed by a semi-colon. Next is a concatenation involving the symbol and certain variables, followed by another semi-colon. Last is a statement containing the same certain variables, followed by an upside-down exclamation point.

The sense of this definition, of course, is that "$\mathbf{P} \subseteq \mathbf{Q}$" may be used to abbreviate "$\forall x(\mathbf{P}[x] \Rightarrow \mathbf{Q}[x])$". Any term in the language may replace **P** or **Q**.

The 𝕊 symbol may be introduced or eliminated. An example where it is introduced:

$\forall x(\mathbf{P}[x] \Rightarrow \mathbf{P}[x])$,! 6 (∀I: 2,5) ¡

$\mathbf{P} \subseteq \mathbf{P}$,! 7 (𝕊I: P1,6) ¡

And a case where it is eliminated:

$\mathbf{P} \subseteq \mathbf{Q}$,! 3 (&E: 2) ¡

$\forall x(\mathbf{P}[x] \Rightarrow \mathbf{Q}[x])$,! 4 (𝕊E: P1,3) ¡

Here the "P" of the "P1" means that the definition appears in the same chapter as the line being justified. If the line occurred in a subsequent chapter of the same section, "C1.1" would have been used (the definition appearing in the first chapter). If it occurred in a subsequent section, "II1.1" would have been used (the definition appearing in the first chapter of section II).

Needless to say, except for different terms replacing **P** or **Q**, and differences in whitespace, the statements must be exactly as appear in the 𝕊 definition for the introduction and elimination rules to be used. For instance, the appearance of parentheses or indeed any additional symbols would prevent the use of elimination. The following is incorrect:

($\mathbf{Q} \subseteq \mathbf{R}$) ,! 7 (&E: 6) ¡

! INCORRECT USE ¡

($\forall x(\mathbf{Q}[x] \Rightarrow \mathbf{R}[x])$) ,! 8 (𝕊E: P1,7) ¡

So is the following (the rule does not apply to sub-statements):

Q ⊆ **R** & **R** ⊆ **Q** ,! 7 (&E: 6) ¡

! INCORRECT USE ¡

∀x(**Q**[x] ⇒ **R**[x]) & ∀x(**R**[x] ⇒ **Q**[x]) ,! 8 (𝕊E: P1,7) ¡

The 𝔻 Definition

Here are two examples of a 𝔻 definition:

! **1.** (**b** _ **c**) represents the finite interval between **b** and **c** (inclusive). ¡

𝔻 _ ; (**b** _ **c**) ; ; {a : ≤[**b**,a] & ≤[a,**c**]} ¡

and

! **1.** ϕ represents the predicate of non-identical things, which of course is empty (nothing satisfies it). ¡

𝔻 ϕ ; ϕ ; ; {a : ¬ a = a} ¡

The new symbol--in the two examples "_" or "ϕ"-- follows "𝔻" and is followed by a semi-colon. Next comes a concatenation involving the symbol and perhaps certain variables. If there are variables, then the term must be enclosed in parentheses, otherwise no. Because neither example involves a pre-supposition (this will be explained in more detail below), two semi-colons follow. Finally comes a term, which must contain the same variables as appeared earlier in the line.

The evident meaning of the first definition is that "(**b** _ **c**)" is to abbreviate "{a : ≤[**b**,a] & ≤[a,**c**]}".

These 𝔻-defined terms may be introduced or eliminated.

An instance of 𝔻I:

∀x ({a : ≤[**b**,a] ∨ ≤[a,**c**]}[x] ⇔ ≤[**b**,x] & ≤[x,**c**])

,! 2 (Pred) ¡

∀x ((**b** _ **c**)[x] ⇔ ≤[**b**,x] & ≤[x,**c**]) ,! 3 (𝔻I: P1,2) ¡

And of 𝔻E:

((**b**+1) _ **c**)[**b**] ,! 2 (Prem) ¡

{a : ≤[(**b**+1),a] & ≤[a,**c**]}[**b**] ,! 3 (𝔻E: P1,2) ¡

Pre-Suppositions

As recently mentioned above, the FOEA language permits pre-suppositions. That is to say, the use of certain terms in an atomic statement may pre-suppose or imply that a condition obtains. For example, the use of "(**x**')" ("the successor of **x**") in an atomic statement implies that **x** is a natural number, i.e. that ω[**x**]. When **x** is not a natural number, "(**x**')" does not refer, and an atomic statement involving the term is not true; for instance, it would not be the case that (**x**') = (**x**'). Indeed, it can be easily proven in FOEA that ¬ ω[**x**] ⇒ ¬ (**x**') = (**x**').

Terms with pre-suppositions impact both the ∀E and ∃I rules. Only terms which refer successfully may be substituted for a universally quantified variable--e.g. one cannot simply assume that (**x**') = (**x**') follows from ∀x x = x; (**x**') = (**x**') may be asserted only if ω[**x**] has already been asserted. Similarly, one may apply ∃I on the term (**x**') only if ω[**x**] has been asserted. For instance, one cannot go from

(¬ ω[**x**] ⇒ ¬ (**x**') = (**x**'))

to deduce

∃x (¬ ω[x] ⇒ ¬ x = x).

The FOEA system may be thought of containing, at any stage, a complete list of terms which refer successfully. For instance, (**x**') is on this list if and only if ω[**x**] has been asserted; if on the list, (**x**') may be substituted for universally quantified variables.

The 𝕋 Definition

The 𝕋 definition is akin to a baptism: by proving that one and only thing exists, we introduce a term to refer to it. In FOEA all instances are such that a condition must be premised in order for the one and only one thing to exist, so all uses involve presuppositions.

For instance, it turns out that we are able to prove the following two propositions:

! **18.** ¡

⊦ ∀R∀x (**f** R & (R^D)[x] ⇒ ∃a R[x,a]) ¡

and

! **19.** ¡

⊦ ∀R∀x (**f** R & (R^D)[x] ⇒ ∀y∀z(R[x,y] & R[x,z] ⇒ y = z)) ¡

The first says that, "if R is a function and x is in its domain, then there exists an a such that R[x,a]", the second that, "if R is a function and x is in its domain, then whenever R[x,y] and R[x,z], y and z must be the same thing." Thus, when **f** R & (R^D)[x] holds, we can be sure there is one and only one thing a such that R[x,a]. We therefore are permitted to introduce a term which refers to this, as such:

! **20.** (**R**´**x**) refers to the thing **y** to which **x** bears **R**. It presupposes that **R** is functional and **x** is in the domain of **R**, which ensures that **y** exists (P18) and is unique (P19). ¡

𝕋 ´ ; (**R**´**x**) ; **f** **R** & ($\mathbf{R}^D$)[**x**] ; **R**[**x**,**y**] ¡! (𝕋D: P18,P19) ¡

A 𝕋 definition has to be justified, and the comment at the end explains that the "𝕋D" rule is being used (in fact, it is the only rule which justifies a 𝕋 definition), based on the two preceding propositions, P18 and P19.

There are the usual associated rules. First, introduction:

f R & (**R**^{D})[**x**] ,! 8 (&I: 3,7) ¡

R[**x**,(**R**´**x**)] ,! 9 (𝕋I: P20,8) ¡

Since (**R**´**x**) is the unique thing y such that **R**[**x**,y] when **f R** & (**R**^{D})[**x**], then **R**[**x**,(**R**´**x**)]. Notice that, previously, the pre-supposition **f R** & (**R**^{D})[**x**] has been asserted.

Secondly, elimination:

(**R**´**x**) = **y** ,! 4 (&E: 2) ¡

f R & (**R**^{D})[**x**] ,! 5 (𝕋E: P20,4) ¡

When an atomic statement has been asserted with (**R**´**x**) in an argument place, then the pre-supposition holds and may be asserted. Remark the requirement about the argument place is strict: (**R**´**x**) may not be a sub-term in an argument place, so for instance {a : a = (**R**´**x**)} appearing in an argument place does not allow one to assert **f R** & (**R**^{D})[**x**].

𝔻 Definitions With Pre-Suppositions

𝔻 definitions also allow presuppositions. For instance, “(**n**')” is introduced to abbreviate the term

“((σ⌈ ω)´**n**)”.

“((σ⌈ ω)´**n**)”, as has just been pointed out in the discussion of 𝕋 definitions, involves the supposition that

f (σ⌈ ω) & ((σ⌈ ω)^{D})[**n**]

Once it has been proven that

! **11.** ¡

⊦ ∀n (ω[n] ⇒ **f** (σ ⌈ ω) & ((σ ⌈ ω)^{D})[n]) ¡

it is therefore possible to make the following definition:

! **12.** ¡

𝔻 ' ; (**n**') ; ω[**n**] ; ((σ ⌈ ω)´**n**) ¡! (𝔻D: III8.20,P11) ¡

Obviously, if ω[**n**] holds, then **f** (σ ⌈ ω) & ((σ ⌈ ω)^{D})[**n**], and so ((σ ⌈ ω)´**n**) makes sense.

Remark that 𝔻 definitions not involving terms with pre-suppositions are not followed by any justification. Here, though, the rule is “𝔻D” and makes reference to “III8.20” and “P11”.

The 𝔻I and 𝔻E rules remains the same, with the exception that for the 𝔻I rule, the pre-supposition must be proven before the term is substituted:

ω[**n**] ,! 2 (Prem) ¡

(ω[**n**] ⇒ **f** (σ ⌈ ω) & ((σ ⌈ ω)^{D})[**n**]) ,! 3 (∀E: P11) ¡

ω[**n**] ⇒ **f** (σ ⌈ ω) & ((σ ⌈ ω)^{D})[**n**] ,! 4 (()E: 3) ¡

f (σ ⌈ ω) & ((σ ⌈ ω)^{D})[**n**] ,! 5 (⇒E: 2,4) ¡

(σ ⌈ ω)[**n**,((σ ⌈ ω)´**n**)] ,! 6 (𝕋I: III8.20,5) ¡

(σ ⌈ ω)[**n**,(**n**')] ,! 7 (𝔻I: P12,2,6) ¡

If the 𝔻 term appears in the argument place of an atomic proposition, then the pre-supposition obtains, using the 𝔻P rule:

(**n**') = **m** ,! 2 (Prem) ¡

ω[**n**] ,! 3 (𝔻P: P12,2) ¡

As before, (**n**') may be substituted for a universally quantified variable, only when ω[**n**] has already be asserted.

In practice, these definition rules involving pre-suppositions are seldom used in the proofs, except for when terms with pre-suppositions are introduced for a universally quantified variable, which happens frequently from Section IV on.

Axioms and Rules of Inference

Axioms and rules of inference are stated in the file I.2, “Language, Axioms and Rules,” which may be found at:

www.andrewboucher.com/papers/foea/I2.pdf

The logical axioms and rules of inference should be natural to anyone with some knowledge of modern logic. The mathematical axioms are those stated in “Systems of Foundations of Arithmetic,” which may be found at:

www.andrewboucher.com/papers/foundations_of_arithmetic.pdf

or in “A Philosophical Introduction to a Foundation of Elementary Arithmetic,” which may be found at

www.andrewboucher.com/papers/foea.htm

Second Version (Dec 2001-Apr 2002)

This is the second version of A Foundation of Elementary Arithmetic. It is distinguished from the first version in several respects. First, the mathematical axioms were changed to the counting axioms, replacing the numbering axioms which are now proved as consequences. Secondly, lines, propositions and definitions are numbered and justifications use these numbers for their references. Thirdly, there has been a certain amount of cleaning, in proofs and in the order of propositions in chapters. And fourthly, there are many additional comments.