UNIVERSITÉ YORK UNIVERSITY

A Basic Formal Equational Predicate Logic

George Tourlakis

Technical Report CS-1998-09

November 25, 1998

Department of Computer Science

4700 Keele Street North York, Ontario M3J 1P3 Canada

A Basic Formal Equational Predicate Logic

George Tourlakis, York University
tgeorge@yorku.ca

Abstract

We present the details of a formalization of Equational Predicate Logic based on a propositional version of the Leibniz rule (PSL—Propositional Strong Leibniz), EQN (equanimity) and TR (transitivity).

All the above rules are "strong", that is, they are applicable to arbitrary premises (not just to absolute theorems).

It is shown that a strong no-capture Leibniz (SLCS—Strong Leibniz with "Contextual Substitution"), and a weak full-capture version (Weak-Leibniz with Uniform Substitution, or WLUS) are derived rules. "Weak" means that the rule is only applicable on absolutely deducible premises.

We also derive general rules MON (monotonicity) and AMON (antimonotonicity), which allow us to "calculate" appropriate conclusions $\vdash C[p \setminus A] \Rightarrow C[p \setminus B]$ or $\vdash C[p \setminus A] \Leftarrow C[p \setminus B]$ from the assumption $\vdash A \Rightarrow B$.

Introduction.

This note builds further on [To], where the logical "calculus" of *Equational (Predicate) Logic* outlined in [GS1] was formalized and shown to be sound and complete.

We propose here a simpler formalization than the one in [To], basing the proof-apparatus solely on *propositional* rules of inference—one of which, of course, is a version of "Leibniz". This entails an unconstrained Deduction Theorem (contrast with [To]), which in turn further simplifies the steps of our reasoning.

While our "foundations" include "just" a propositional version of Leibniz, we show that there are derived rules valid in the logic, which allow the use of Leibniz-style substitution within the scope of a quantifier.

We also address one "weakness"—to which David Gries has already called attention in [Gr]—of the current literature ([DSc, GS1]) on equational or calculational reasoning. That is, while it is customary to mix =-steps (that is, an application of a *conjunctional* $\equiv$) and $\Rightarrow$-steps (that is, an application of a *conjunctional* $\Rightarrow$) in a calculational proof, and while we have well documented rules to handle the former, yet the latter type of step normally seems to rely on a compendium of *ad hoc* rules. We hope to have contributed towards remedying this state of affairs, as we present a unifying, yet simple and rigorous way to understand, ascertain validity, and therefore annotate and utilize $\Rightarrow$-steps, using the rules *monotonicity* and *antimonotonicity*.

We conclude with a section on soundness and completeness of the proposed Logic.

The term "basic" in the title is meant to convey that we include no more than what is necessary to lay the foundations. In particular, examples that illustrate the power of calculational reasoning were left out.

The layout of the paper is as follows: Section 1 introduces the formal language. Section 2 the axioms, the rules of inference, and sets the rules of the game (definitions of the two main types of substitution,[†] of theorem and of proof). Section 3 introduces a few metatheorems, including the Deduction Theorem. The "main lemma" in section 6 is lemma 6.15 that shows the eliminability of propositional variables. An Appendix argues that all the axioms in [GS1] are here theorems, but the exposition in the Appendix is no more than a "link" to [To].

1. Syntax

Equational (first order) logic, like all (first order) logic, is "spoken" in a *first order language*, L. L is a triple $(V, \mathbf{Term}, \mathbf{Wff})$, where V is the *alphabet*, i.e., the set of basic syntactic objects (symbols) that we use to built "terms" and "formulas". We start with a description of V, and then we describe the set of terms (**Term**) and the set of formulas (**Wff**).

Alphabet

1. *Object variables.* An enumerable set $x_1, x_2, \ldots$. We normally use the metasymbols x, y, z, u, v, w with or without primes to stand for (object) variables.

2. *Boolean* (or *Sentential*, or *Propositional*) *variables.*

An enumerable set $v_1, v_2, \ldots$. We normally use the metasymbols p, q, r with or without primes to stand for Boolean variables.

These variables will be *only* used to give a user-friendly notation for the various versions of the rule Leibniz.

3. Equality (between "objects"—see 1.3 below for its syntactic role), "$\approx$".[‡]

4. Brackets, (and).

5. The Boolean or propositional connectives, $\equiv, \Rightarrow, \vee, \wedge, \neg$.

6. The quantifier, $\forall$ (the existential quantifier, $\exists$, will be taken to be a metasymbol, introduced definitionally).

[†] There is a significant departure from [To] in the notation for substitution. In [To] there was just one notation—using the symbol $[* := **]$, where $*$ is a variable and $**$ a formula or a term. This notation would be annotated by surrounding text, which would indicate if capture of free variables was, or was not, allowed. Here, instead, we ask the notation to fend for its meaning, using different notations for the capture and no capture versions.

[‡] Following the practice in Enderton [En], we use $\approx$ for *formal* equality and $=$ for informal (metamathematical) equality. This will enable us to write in the metatheory, for example, $A = B$ where A and B are formulas, meaning that the "strings" A and B are identical. A conflict will arise though, since we will be also using $=$ quasi-formally (in equational style proofs) as a *conjunctional* alias of $\equiv$.

7. The special Boolean (propositional) constants *true* and *false*.

8. A set of symbols (possibly empty) for *constants*. We normally use the meta-symbols a, b, c, d, e, with or without subscripts, to stand for constants unless we have in mind some alternative "standard" notation in selected areas of application of the 1st order logic (e.g., $\emptyset$, 0, ω, etc.).

9. A set of symbols for *predicates* or *relations* (possibly empty) for each possible "arity" $n > 0$. We normally use P, Q, R with or without primes to stand for predicate symbols.

10. Finally, a set of symbols for *functions* (possibly empty) for each possible "arity" $n > 0$. We normally use f, g, h with or without primes to stand for function symbols.

1.1 Remark. *Any two* symbols mentioned in items 1–10 are distinct. Moreover (if they are built from simpler "sub-symbols", e.g., $x_1, x_2, x_3, \ldots$ might *really* be $x|x, x||x, x|||x, \ldots$), none is a *substring* (or *subexpression*) of any other.

1.2 Definition. (*Terms*) The set of *terms*, **Term**, is the $\subseteq$-*smallest* set of strings or "expressions" over the alphabet 1–10 with the following two properties:

Any of the items in 1 or 8 (a, b, c, x, y, z, etc.) are in **Term**.

If f is a function[†] of arity n and $t_1, t_2, \ldots, t_n$ are in **Term**, then so is the string $ft_1t_2 \ldots t_n$. □

1.3 Definition. (*Atomic Formulas*) The set of *atomic formulas*, **Af**, contains precisely:

1) The symbols *true*, *false*, and every Boolean variable (that is, $p, q, \ldots$).

2) The strings $t \approx s$ for every possible choice of terms t, s.

3) The strings $Pt_1t_2 \ldots t_n$ for every possible choices of n-ary predicates P (for all choices of $n > 0$) and all possible choices of terms $t_1, t_2, \ldots, t_n$.

□

[†] We will omit the qualification "symbol" from terminology such as "function symbol", "constant symbol", "predicate symbol".

1.4 Definition. (*Well-Formed Formulas*) The set of *well-formed formulas*, **Wff**, is the $\subseteq$-*smallest* set of strings or "expressions" over the alphabet 1–10 with the following properties:

a) $\mathbf{Af} \subseteq \mathbf{Wff}$.

b) If A, B are in **Wff**, then so are $(A \equiv B)$, $(A \Rightarrow B)$, $(A \wedge B)$, $(A \vee B)$.

c) If A is in **Wff**, then so is $(\neg A)$.

d) If A is in **Wff** and x is any object variable (*which may or may not occur (as a* substring*) in the formula* A), then the string $((\forall x)A)$ is also in **Wff**.

We say that A is the *scope* of $(\forall x)$.

□

1.5 Remark. (1) A, B in the definition are so-called *syntactic* or meta-variables, used as *names* for (arbitrary) formulas. In general, we will let the letters A, B, C, D, E (with or without primes) be names for well-formed formulas, or just *formulas* as we often say. The definition of **Wff** given above is standard. In particular, it allows formulas such as $((\forall x)((\forall x)x = 0))$ in the interest of making the formation rules "context free" (in some presentations, formation rule 1.4(d) requires that x be not already quantified in A).

(2) We introduce a meta-symbol $(\exists)$ *solely in the metalanguage* via the definition "$((\exists x)A)$ stands for, or abbreviates, $(\neg((\forall x)(\neg A)))$."

(3) We often write more explicitly, $((\forall x)A[x])$ and $((\exists x)A[x])$ for $((\forall x)A)$ and $((\exists x)A)$. This is intended to draw attention to the variable x of A, which has now become "bound". Of course, notwithstanding the notation $A[x]$ (which only says that x *may* occur in A), x might actually not be a substring of A. In that case, intuitively, $((\forall x)A)$, $((\exists x)A)$ and A "mean" the same thing. This intuition is actually captured by 3.11.

(4) To minimize the use of brackets in the metanotation we adopt standard priorities, that is, $\forall$, $\exists$, and $\neg$ have the highest, and then we have (in decreasing order of priority) $\wedge, \vee, \Rightarrow, \equiv$. All associativities are *right* (this is in variance with [GS1], but is just another acceptable—and common—convention of how to be sloppy in the metalanguage, and get away with it).

(5) The language just defined, L, is *one-sorted*, that is, it has a single *sort* or *type* of object variable. The reasons for this choice are articulated in [To] and we will not repeat them here.

A variable that is quantified is *bound in the scope of the quantifier*. Non quantified variables are *free*. The above loose description is tightened below by induction on formulas.

1.6 Definition. (*Free and bound variables*) An object variable x occurs free in a term t or atomic formula A iff it occurs in t or A as a substring.

x occurs free in $(\neg A)$ iff it occurs free in A.

x occurs free in $(A \circ B)$—where $\circ \in \{\equiv, \vee, \wedge, \Rightarrow\}$—iff it occurs free in at least one of A or B.

x occurs free in $((\forall y)A)$ iff x occurs free in A, *and* $y \neq x$.[†]

The y in in $((\forall y)A)$ is, of course, *not* free—even if it is so in A—as we have just concluded in this inductive definition. We say that it is *bound* in $((\forall y)A)$. Trivially, terms and atomic formulas have no *bound* variables. □

1.7 Definition. (*Closed terms and formulas. Sentences. Open formulas*) A term or formula is *closed* iff no free variables occur in it. A closed formula is called a *sentence*.

A formula is *open* iff it contains no quantifiers (thus, it could also be closed!) □

2. Axioms and Rules of Inference

Now that we have our language, L, we will embark on using it to formally effect "deductions". These deductions "start" at the "axioms". Deductions employ "acceptable" purely syntactic—i.e., based on *form*, not on *substance*—rules that allow us to write a formula down (to deduce it) solely because certain other formulas *that are syntactically related to it* were already deduced (i.e., written down). These string-manipulation rules are called "rules of inference". We describe in this section the axioms and the rules of inference that we will accept into our logical calculus.

These are chosen so that a logic equivalent to that in [Bou, En] results. The characterizing feature will be that *all primary rules of inference* are "propositional". This will entail a very simple version of the Deduction Theorem that is applicable without constraints. We have the choice of taking all tautologies as schemata in **Ax1** below, or restricting the set to just those axiom schemata given in [GS3].

We will succumb to the temptation of taking the big leap and adopting *all* tautologies as axioms. Technically, this is fine, since the set of all tautologies is "recognizable" (recursive),[‡] and all tautologies will have to be deducible at any rate, no matter how we start-up the logic. Pedagogically it is also fine. At this point the student presumably knows of the *Tautology Theorem* of Post (completeness of Propositional calculus)—perhaps even of its proof—and she or he would rather navigate through the challenges of the new calculus unimpeded by a requirement to re-discover (prove) from scratch all the tautologies he or

---

[†] Recall that $=$ and $\neq$ are used in the metalanguage as equality and inequality (respectively). In this case, we are comparing the strings x and y.

[‡] It is common practice that those axioms of logic that are "common" to all mathematics (logical) form a "recognizable" set.

she needs as tools here, all over again. We will need a precise definition of tautologies in our first order language L.

2.1 Definition. (*Prime formulas in* **Wff**) A formula $A \in$ **Wff** is a *prime formula* iff it is any of

Pri1. Atomic

Pri2. A formula of the form $(\forall x)A$.

Let $\mathcal{P}$ denote the set of *all* prime formulas in our language. Clearly, $\mathcal{P}$ contains each propositional variable $v_1, v_2, \ldots$. $\square$

That is, a prime formula has no "explicit" propositional connectives (in the case labeled **Pri2** any propositional connectives are hidden inside the scope of $(\forall x)$).

Clearly, $A \in$ **Wff** iff A is a *Propositional Calculus formula over* $\mathcal{P}$ (i.e, propositional variables will be all the strings in $\mathcal{P} - \{true, false\}$).

2.2 Definition. (*Tautologies in* **Wff**) A formula $A \in$ **Wff** is a *tautology* iff it is so when viewed as a Propositional Calculus formula over $\mathcal{P}$.

We call the set of all tautologies, as defined here, **Taut**. The symbol $\models_{\mathbf{Taut}} A$ says $A \in$ **Taut**. $\square$

The following generalizes 2.2 and will also be needed.

2.3 Definition. (*Tautologically Implies, for formulas in* **Wff**) Given formulas $B \in$ **Wff** and $A_i \in$ **Wff**, for $i = 1, \ldots, m$. We say that $A_1, \ldots, A_m$ *tautologically implies* B, in symbols $A_1, \ldots, A_m \models_{\mathbf{Taut}} B$, iff (viewing the formulas as Propositional Calculus formulas over $\mathcal{P}$) *every truth assignment* to the prime formulas that makes each of the A_i true, also makes B true. $\square$

While a definition for an infinite set of premises is possible, we will not need it here.

Before presenting the axioms, we introduce some notational conventions regarding *substitution*.

2.4 Definition. (*Substitutions*) We will have two type of substitutions:

Contextual Substitution: $A[p := W]$ and $A[x := t]$ denote, respectively, the result of replacing *all* p by the formula W and *all free* x by the term t, *provided no variable of* W *or* t *was "captured" (by a quantifier) during substitution.* If the proviso is not valid, then the substitution is undefined.

Uniform Substitution: $A[p \setminus W]$ and $A[x \setminus t]$ denote, respectively, the result of replacing *all* p by the formula W and *all free* x by the term t. *No restrictions.*

The symbols $[p := *]$ and $[p \setminus *]$ above, where $*$ is a W or a t, lie in the metalanguage. These metasymbols have the highest priority. $\square$

2.5 Remark. (1) An inductive definition (by induction on terms and formulas) of the string $A[x := t]$ is instructive and is given below:

First off, let us define $s[x := t]$, where s is also a term:

If $s = x$[†], then $s[x := t] = t$. If $s = a$ (a constant), then $s[x := t] = a$. If $s = y$ and $y \neq x$ (i.e., they are different strings!), then $s[x := t] = y$.

If $s = fr_1 \ldots r_n$—where f has arity n and $r_1, \ldots, r_n$ are terms—then $s[x := t] = fr_1[x := t]r_2[x := t] \ldots r_n[x := t]$.

We turn now to formulas.

If A is *true*, *false* or p (a boolean variable), then $A[x := t] = A$. If $A = s \approx r$, where s and r are terms, then $A[x := t] = s[x := t] \approx r[x := t]$. If $A = Pr_1 \ldots r_n$ (P has arity n), then $A[x := t] = Pr_1[x := t]r_2[x := t] \ldots r_n[x := t]$.

If $A = (B \circ C)$, where $\circ \in \{\equiv, \vee, \wedge, \Rightarrow\}$, then $A[x := t] = (B[x := t] \circ C[x := t])$.

If $A = (\neg B)$, then $A[x := t] = (\neg B[x := t])$.

In both cases above, the left hand side is defined just in case the right hand side is.

Finally, (the "interesting case"): say $A = ((\forall y)B)$. If $y = x$, then (x is not free in A) $A[x := t] = A$.

If $y \neq x$ and $B[x := t]$ is defined, then $A[x := t]$ is defined *provided y is not a substring of t*. In that case, $A[x := t] = ((\forall y)B[x := t])$.

(2) Similarly, we define $A[p := W]$ inductively below ($\circ \in \{\equiv, \vee, \&, \Rightarrow\}$ as before):

$$A[p := W] = \begin{cases} W & \text{if } A = p \\ A & \text{if } A \text{ is atomic, but not } p \\ (\neg B[p := W]) & \text{if } A = (\neg B) \text{ and } B[p := W] \text{ is defined} \\ (B[p := W] \circ C[p := W]) & \text{if } A = (B \circ C) \text{ and } B[p := W] \textit{ and } \\ & \qquad C[p := W] \text{ are defined} \\ ((\forall y)B[p := W]) & \text{if } A = ((\forall y)B), \text{ provided } B[p := W] \\ & \quad \text{is defined } \textit{and } y \text{ is not free in } W \end{cases}$$

The cases for $A[x \setminus t]$ and $A[p \setminus W]$ have exactly the same inductive definitions, *except* that we drop the hedging "if defined" throughout, and we also drop the restriction that y (of $(\forall y)$) be not free in W or t.

2.6 Definition. (*Axioms and Axiom schemata*) The axioms (*schemata*) are *all* the possible "partial" generalizations[‡] of the following (exactly as in [En][§]):

[†]For one last time, recall that $=$ is metalogical, and it here denotes equal strings!

[‡]B is a *partial generalization* of A iff it is the expression consisting of A, prefixed with zero or more strings $(\forall x)$—x may or may not occur free in A. Repetitions of the same prefix-string $(\forall x)$ are allowed. The well known "universal closure of A", that is $(\forall x_1)(\forall x_2) \ldots (\forall x_n)A$—where $x_1, x_2, \ldots, x_n$ is the full list of free variables in A—is a special case.

[§]Actually, [En] only allows atomic formulas in **Ax6** below, and derives **Ax6** as a theorem. We chose this avenue for convenience.

Ax1. All formulas in **Taut**.

Ax2. (*Schema*) For every formula A,

$$(\forall x)A \Rightarrow A[x := t], \text{ for any term } t.$$

By 2.4, *the notation imposes a condition on t.* The condition is that during substitution *no* variable of t (all such are free, of course) was "captured" by a quantifier. We say that "t must be substitutable in x".

NB. We often see the above written (in metalinguistic *argot*) as

$$(\forall x)A[x] \Rightarrow A[t]$$

or even

$$(\forall x)A \Rightarrow A[t]$$

where the presence of $A[x]$ (or $(\forall x)A$, or $(\exists x)A$) and $A[t]$ in the *same context* means that t *replaces contextually* all x occurrences in A.

Ax3. (*Schema*) For every formula A and variable x *not free in* A,

$$A \Rightarrow (\forall x)A.$$

Ax4. (*Schema*) For every formulas A and B,

$$(\forall x)(A \Rightarrow B) \Rightarrow (\forall x) \Rightarrow A(\forall x)B$$

Ax5. (*Schema*) For *each* object variable x, the formula $x \approx x$.

Ax6. (*Leibniz's characterization of equality—1st order version. Schema*) For any formula A, any object variable x and any term t, the formula

$$x \approx t \Rightarrow A \equiv A[x := t].$$

NB. The above is written usually as

$$x \approx t \Rightarrow A[x] \equiv A[t]$$

or even

$$x \approx t \Rightarrow A \equiv A[t]$$

as long as we remember that the substitution of t for x must be contextual.

□

2.7 Remark. (1) In any formal setting that introduces many-sorts explicitly in the syntax, one will need as many versions of **Ax2**–**Ax6** as there are sorts.

(2) Axioms **Ax5**–**Ax6** characterize equality between "objects". Adding these two axioms makes the logical system (explicitly) applicable to mathematical theories such as number theory and set theory. These axioms will be used to prove the "one point rule" of [GS1] (in the Appendix).

(3) In **Ax2** and **Ax6** we imposed the condition that t must to be "substitutable" in x by utilizing contextual substitution $[x := t]$.

Here is why:

Take A to stand for $(\exists y)\neg x \approx y$. Then $(\forall x)A \Rightarrow A[x \setminus y]$ is

$$(\forall x)(\exists y)\neg x \approx y \Rightarrow (\exists y)\neg y \approx y$$

and $x \approx y \Rightarrow A \equiv A[x \setminus y]$ is

$$x \approx y \Rightarrow (\exists y)\neg x \approx y \equiv (\exists y)\neg y \approx y$$

neither of which, obviously, is universally valid.

The meta-remedy is to move the quantified variable(s) out of harm's way, i.e., rename them so that no quantified variable in A has the same name as any (free, of course) variable in t.

This renaming is formally correct (i.e., it does not change the meaning of the formula) as we will see in the "variant" (meta)theorem (3.10). Of course, it is always possible to effect this renaming since we have countably many variables, and only finitely many appear free in t *and* A.

This trivial remedy allows us to render the conditions in **Ax2** and **Ax6** harmless. Essentially, a t is always "substitutable" (so that we can use $[x \setminus t]$ instead of the restrictive $[x := t]$) *after renaming.*

2.8 Definition. (*Rules of Inference*) The following three are the *rules of inference.* These rules are relations on the set **Wff** and are written traditionally as "fractions". We call the "numerator" the *premise(s)* and the "denominator" the *conclusion.*

We say that a rule of inference is *applied* to the formula(s) in the numerator, and that it *yields* (or *results in*) the formula in the denominator. We emphasize that the domain of the rules we describe below is the set **Wff**. That is why we call the rules "strong" (a "weak" rule applies on a proper subset of **Wff** only. That subset is not yet defined(!). No wonder then that we prefer "strong" over "weak" rules).

Any set $S \subseteq$ **Wff** is *closed under* some rule of inference iff whenever the rule is applied to formulas in S, it also yields formulas in S.

Inf1. (*Propositional (Strong) Leibniz, PSL*) For any formulas A, B, C and any propositional variable p (which may or may not occur in C)

$$\frac{A \equiv B}{C[p := A] \equiv C[p := B]} \qquad (PSL)$$

on the condition that p is *not* in the scope of a quantifier.

Given the condition on p, *it makes no difference* if in PSL above we used $[p \setminus *]$ instead of $[p := *]$.

Inf2. (*Equanimity, EQN*) For any formulas A, B

$$\frac{A, A \equiv B}{B}$$

Inf3. (*Transitivity of* $\equiv$*, TR*) For any formulas A, B, C

$$\frac{A \equiv B, B \equiv C}{A \equiv C}$$

$\square$

2.9 Remark. (1) PSL is the primary rule in the propositional calculus fragment of Equational Logic as it is presented in [GS1]. An additional predicate calculus version is also given there (twin rule 8.12; the second of the two needs a correction—see the Appendix). The Leibniz rule (or its variants) is at the heart of equational or calculational reasoning. In standard approaches to logic it is not a primary rule, rather it appears as the well known "derived rule" (metatheorem) that if $\Gamma \vdash A \equiv B$[†] and if we replace one or more occurrences of the subformula A of a formula D (here D is $C[p := A]$) by B, to obtain D' (that is $C[p := B]$), then $\Gamma \vdash D \equiv D'$. *No restriction on p is necessary* (as we prove in section 4). I.e., we show that the above quoted "Leibniz" is a derived rule in our system.

Shoenfield [Sh] calls this derived rule "the equivalence theorem".[‡]

(2) [GS1] use "=" for "$\equiv$" in contexts where they want the symbol to act *conjunctionally*, rather than *associatively*, e.g., in successive steps of an equational-style proof. We will follow this practice as well.

This may create a few confusing moments, as we use = in the metalanguage as well!

We next define Γ-theorems, that is, formulas we can *prove from* the *set* of formulas Γ (this Γ may be empty).

2.10 Definition. (Γ-*theorems*) The set of Γ-*theorems*, $\mathbf{Thm}_\Gamma$, is the $\subseteq$-*smallest* subset of **Wff** that satisfies the following:

Th1. $\mathbf{Thm}_\Gamma$ contains as a subset all the axioms defined in 2.6.

[†] The meaning of the symbol $\vdash$ is defined in 2.10 below.

[‡] The syntactic apparatus in [Sh], but *not ours*!—see section 4—allows a stronger "Leibniz". It allows **Inf1** *with* uniform substitution and *without* the caveat on p! See also [To].

We call these formulas *the logical axioms.*

Th2. $\Gamma \subseteq \mathbf{Thm}_\Gamma$.

We call every member of Γ a *nonlogical axiom.*

Th3. $\mathbf{Thm}_\Gamma$ is *closed under* each rule **Inf1**–**Inf3**.

The (meta)statement $A \in \mathbf{Thm}_\Gamma$ is traditionally written as $\Gamma \vdash A$, and we say that A *is proved from* Γ or that it is a Γ*-theorem.*

If $\Gamma = \emptyset$, then rather than $\emptyset \vdash A$ we write $\vdash A$. We often say in this case that A is *absolutely provable* (or provable with no nonlogical axioms).

We often write $A, B, \ldots, D \vdash E$ for $\{A, B, \ldots, D\} \vdash E$. □

2.11 Remark. Now we can spell out what a "weak" rule of inference is: It is a rule whose domain is restricted to be $\mathbf{Thm}_\emptyset$.

None of **Inf1**–**Inf3** is weak.

2.12 Definition. (Γ*-proofs*) A finite sequence $A_1, \ldots, A_n$ of members of **Wff** is a Γ*-proof* iff every A_i, for $i = 1, \ldots, n$ is one of

Pr1. A logical axiom (as in **Th1** above).

Pr2. A member of Γ.

Pr3. The result of a rule **Inf1**–**Inf3** applied to (an) appropriate formula(s) A_j with $j < i$.

□

2.13 Remark. (1) It is a well known result on inductive definitions that $\Gamma \vdash A$ is equivalent to "A appears in some Γ-proof"—in the sense of the above definition—and also equivalent to "A is at the end of some Γ-proof".

(2) It follows from 2.12 that if each of $A_1, \ldots, A_n$ has a Γ-proof and B has an $\{A_1, \ldots, A_n\}$-proof, then B has a Γ-proof. Indeed, simply concatenate each of the given Γ-proofs (in any sequence). Append to the right of that sequence the given $\{A_1, \ldots, A_n\}$-proof (that ends with B). Clearly, the entire sequence is a Γ-proof, and ends with B.

We refer to this phenomenon as the *transitivity of* $\vdash$.

(3) If $\Gamma \subseteq \Delta$ and $\Gamma \vdash A$, then also $\Delta \vdash A$ as it follows from 2.10 or 2.12. In particular, $\vdash A$ implies $\Gamma \vdash A$ for any Γ.

(4) The inductive definition of theorems (2.10) allows one to prove properties of $\mathbf{Thm}_\Gamma$ by *induction on theorems.* The equivalent *iterative definition* 2.12, via the concept of proof, allows us a different kind of induction, on the length of proofs.

3. Basic metatheorems and theorems

3.1 Metatheorem. (Redundant *true*) *For any formula A and any set of formulas Γ, $\Gamma \vdash A$ iff $\Gamma \vdash A \equiv true$.*

Proof.

$$
\begin{array}{rl}
 & A \\
= & \langle A \equiv A \equiv \mathit{true} \text{ is a tautology, hence a logical axiom} \rangle \\
 & A \equiv \mathit{true}
\end{array}
$$

So $\Gamma \vdash A$ iff $\Gamma \vdash A \equiv \mathit{true}$ by EQN.† □

3.2 Metatheorem. (Modus ponens—derived rule) *$A, A \Rightarrow B \vdash B$ for any formulas A and B.*

From 2.10 follows that any of the primary rules can be written "linearly", that is, premises first, followed by $\vdash$—instead of the "fraction-line"—followed by the conclusion.

We (almost) always use this linear format for *derived* rules.

Proof. We have

$$A \quad \langle \text{nonlogical assumption} \rangle \tag{1}$$

and

$$A \Rightarrow B \quad \langle \text{nonlogical assumption} \rangle \tag{2}$$

Thus,

$$
\begin{array}{rl}
 & A \Rightarrow B \\
= & \langle \text{redundant true via (1); plus PSL} \rangle \\
 & \mathit{true} \Rightarrow B \\
= & \langle (\mathit{true} \Rightarrow B) \equiv B \text{ is a tautology} \rangle \\
 & B
\end{array}
$$

Since $A, A \Rightarrow B \vdash A \Rightarrow B$, the above "calculation" and EQN yield $A, A \Rightarrow B \vdash B$. □

Let us call our logic, that is, a language L along with the adopted axioms, rules of inference and the definition (2.10) of Γ-theorems (an) *E-logic* ("E" for Equational).

Let us call *En-logic* what we obtain by *keeping all else the same*, but *adopting modus ponens as the only primary rule of inference*. This is, essentially, the logic in [En] (except that [En] allows neither propositional variables nor propositional constants).

† We take symmetry of $\equiv$ for granted, and leave it unmentioned, due to axiom group **Ax1**.

We may subscript the symbol $\vdash$ by an E or En to indicate in which logic we are working. Thus, e.g., $\Gamma \vdash_{En} A$ means we deduced A from Γ working in the En-logic.

Why introduce En-logic? Its metatheory is a bit simpler, due to the presence of just one solitary rule of inference. For this reason we will often work in En-logic to establish metatheorems, e.g., (below) that we have closure of theorems under $(\forall x)$ (under some reasonable restrictions), Deduction Theorem, etc.

First we need to show that we are not barking up the wrong tree. E-logic and En-logic are equivalent.

3.3 Lemma. (The extended tautology theorem) *If* $A_1, \ldots, A_n \models_{\mathbf{Taut}} B$ *then,* $A_1, \ldots, A_n \vdash B$ *in either E-logic or En-logic.*

Proof. The assumption yields that

$$\models_{\mathbf{Taut}} A_1 \Rightarrow \ldots \Rightarrow A_n \Rightarrow B \tag{1}$$

Thus (the formula in (1) is an axiom of both logics),

$$A_1, \ldots, A_n \vdash A_1 \Rightarrow \ldots \Rightarrow A_n \Rightarrow B \tag{2}$$

Applying modus ponens to (2), n times, we deduce B. □

3.4 Metatheorem. *For any* Γ *and any formula* A, $\Gamma \vdash_{En} A$ *iff* $\Gamma \vdash_E A$.

Proof. We already have shown that modus ponens is a derived rule of E-logic. Thus, if $\Gamma \vdash_{En} A$, then $\Gamma \vdash_E A$.

Conversely, since every rule **Inf1**–**Inf3** of E-logic is *derived* in En-logic, whenever $\Gamma \vdash_E A$ we also get $\Gamma \vdash_{En} A$.

Now, why are **Inf1**–**Inf3** derived in En-logic?

The reason is that each has the form of a tautological implication,

$$A_1, \ldots, A_n \models_{\mathbf{Taut}} B$$

for $n = 1$ (PSL) or $n = 2$ (EQN, TR).[†] □

3.5 Metatheorem. (Generalization) *For any* Γ *and any* A*, if* $\Gamma \vdash A$ *with a* restriction*, then* $\Gamma \vdash (\forall x)A$.

The restriction is: There is a Γ*-proof of* A*, such that* x *does not occur free in any formula of* Γ *that was used in the proof.*

Proof. The (meta)proof will be by induction on the length of a Γ-proof that deduces A while respecting the restriction on x. In view of 3.4, we do the (meta)proof about En-logic (see [En][‡]).

[†] That is why the rules, in particular, PSL, are called "propositional".

[‡] Actually Enderton requires an unnecessarily strong condition: That x is not free in *any* formula in Γ. He does so, presumably, because he offers a proof by induction on Γ-theorems. Induction on Γ-proofs, as we opted for here, is satisfied with a lesser restriction, imposed on *finitely many* formulas of Γ.

The idea is very simple: Let $B_1, B_2, \ldots, B_n$ be such a Γ-proof effected in En-logic, where $B_n = A$. We show (by induction on n), that in the sequence

$$(\forall x)B_1, (\forall x)B_2, \ldots, (\forall x)B_n \tag{1}$$

every formula is a Γ-theorem of En-logic, hence also of E-logic.

Basis $n = 1$. If $B_1 \in \Gamma$, then x is not free in B_1. By **Ax3**, and modus ponens, $B_1 \vdash (\forall x)B_1$, hence $\Gamma \vdash (\forall x)B_1$ by transitivity of $\vdash$.

If B_1 is logical, then $(\forall x)B_1$ is also logical (partial generalization—recall 2.6). Thus, again, $\Gamma \vdash (\forall x)B_1$ (by 2.10 or 2.12).

Assume the claim for $n \leq k$ (Induction Hypothesis, in short I.H.).

We look at $n = k + 1$. If B_n is logical or in Γ, we already have seen how to handle it. Suppose then that B_n is actually there because we had applied modus ponens, namely, $B_i, B_i \Rightarrow B_n \vdash B_n$, and that $B_i \Rightarrow B_n$ is the formula B_j. Of course, i and j are each less than n, hence $\leq k$, so the I.H. applies.

Thus,

$$\Gamma \vdash (\forall x)B_i \tag{2}$$

and

$$\Gamma \vdash (\forall x)(B_i \Rightarrow B_n) \tag{3}$$

Applying modus ponens—via (2) and (3)—twice to the following instance of **Ax4**, $(\forall x)(B_i \Rightarrow B_n) \Rightarrow (\forall x)B_i \Rightarrow (\forall x)B_n$, we get $\Gamma \vdash (\forall x)B_n$. $\square$

An important observation flows immediately from the proof of 3.5.

The sequence (1) can be "padded" to be a Γ-proof *without using any additional Γ-formulas beyond those used to derive A in the first place.* Indeed, inspect the Basis and Induction Steps: They utilized whatever was utilized out of Γ already, modus ponens and logical axioms, in order to show that each $(\forall x)B_i$ is deducible.

3.6 Corollary. ("Weak" Generalization) *For any formula A, if $\vdash A$, then $\vdash (\forall x)A$.*

Proof. Take $\Gamma = \emptyset$ above. $\square$

We trivially have a derived rule *specialization*, sort of the converse of generalization. It says that if $\Gamma \vdash (\forall x)A$, then $\Gamma \vdash A$. To see that it holds, note that $(\forall x)A \Rightarrow A[x := x]$ is a logical axiom (it is easy to check that x is always substitutable in x). By modus ponens, $\Gamma \vdash A$ (of course, $A[x := x] = A$).

In particular, we can state: $\vdash A$ iff $\vdash (\forall x)A$.

"Weak" implies that there is a strong generalization rule as well. That is the "rule" $A \vdash (\forall x)A$.

This rule is *not* derivable in E-logic. We will see why once we have proved the Deduction Theorem.

3.7 Corollary. (Substitution of terms) *If there is a* Γ*-proof of* $A[x_1, \ldots, x_n]$ *so that none of the variables* $x_1, \ldots, x_n$ *occurs free in the* Γ*-formulas used in the proof, then* $\Gamma \vdash A[t_1, \ldots, t_n]$*, for any terms* $t_1, \ldots, t_n$*.*

Proof. Of course, the t_i must be "substitutable" in the respective variables. One can comfortably be silent about this in view of the variant theorem (3.10, below).

We illustrate the proof for $n = 2$. What makes it interesting is the requirement to have *simultaneous substitution.* To that end we first substitute into x_1 and x_2 *new* variables z, w—i.e., not occurring in the t_i nor in A (neither free nor bound).

So let $\Gamma \vdash A[x_1, x_2]$ as restricted in the corollary statement.

The proof is the following sequence.

$\langle 3.5 \rangle$

$(\forall x_1)A[x_1, x_2]$

$\langle$**Ax2** and modus ponens; $x_1 := z\rangle$

$A[z, x_2]$

$\langle$3.5—see also the remark following 3.5$\rangle$

$(\forall x_2)A[z, x_2]$

$\langle$**Ax2** and modus ponens; $x_2 := w\rangle$

$A[z, w]$

$\langle$Now $z := t_1, w := t_2$, *in any order,*
is the same as "simultaneous substitution"$\rangle$

$\langle$3.5—see also the remark following 3.5$\rangle$

$(\forall z)A[z, w]$

$\langle$**Ax2** and modus ponens; $z := t_1\rangle$

$A[t_1, w]$

$\langle$3.5—see also the remark following 3.5$\rangle$

$(\forall w)A[t_1, w]$

$\langle$**Ax2** and modus ponens; $w := t_2\rangle$

$A[t_1, t_2]$

□

3.8 Metatheorem. (The Deduction Theorem) *For any formulas* A *and* B *and set of formulas* Γ*, if* $\Gamma, A \vdash B$*, then* $\Gamma \vdash A \Rightarrow B$*.*

NB. Γ, A means $\Gamma \cup \{A\}$. A converse of the metatheorem is also true trivially: That is, $\Gamma \vdash A \Rightarrow B$ implies $\Gamma, A \vdash B$. This follows by modus ponens.

Proof. The proof is by induction on Γ, A-theorems and, once again, it is carried out in En-logic.

Basis. Let B be logical or nonlogical. Then $\Gamma \vdash B$ (2.10).

Trivially, $B \models_{\mathbf{Taut}} A \Rightarrow B$, hence, by 3.3 and the transitivity of $\vdash$, $\Gamma \vdash A \Rightarrow B$.

If B is the same string as A, then $A \Rightarrow B$ is a logical axiom (tautology), hence $\Gamma \vdash A \Rightarrow B$ (2.10).

There is only one induction step.

Modus ponens. Let $\Gamma, A \vdash C$, and $\Gamma, A \vdash C \Rightarrow B$.

By I.H., $\Gamma \vdash A \Rightarrow C$ and $\Gamma \vdash A \Rightarrow C \Rightarrow B$.

Since $A \Rightarrow C, A \Rightarrow C \Rightarrow B \models_{\mathbf{Taut}} A \Rightarrow B$, we have $\Gamma \vdash A \Rightarrow B$. $\square$

3.9 Remark. (1) We now see why our E-logic (equivalently, En-logic) does *not* support strong generalization $A \vdash (\forall x)A$. If it did, then, by the Deduction Theorem that we have just proved,

$$\vdash A \Rightarrow (\forall x)A \tag{i}$$

Even though we have not discussed semantics yet (we do so in section 6), still we can see intuitively that no self-respecting logic should have the above formula as an absolute theorem, since it is not an "absolute truth". For example, over the natural numbers, $\mathbb{N}$, we have an obviously invalid "special case" of the schema (i):

$$x \approx 0 \Rightarrow (\forall x)x \approx 0$$

In some expositions the Deduction Theorem is constrained by requiring that A be closed ([Sh, To]), or a complicated condition on its variables is given ([Men]).

Which version is right? They both are. If all the *primary* rules of inference are "propositional" as they are here, then the Deduction theorem is unconstrained because we do not have strong generalization. If, on the other hand, the rules of inference manipulate object variables via quantification (e.g., strong generalization, or other "stronger" rules are present—see [Men, Sh, To]), then one cannot avoid constraining the application of the deduction Theorem, lest one wants to derive (the invalid) (i) above.

(2) This divergence of approach in choosing rules of inference has some additional repercussions.

One has to be careful in defining the semantic counterpart of $\vdash$, namely, $\models$ (see section 6). One wants the two symbols to "track each other" faithfully (Gödel's completeness theorem).[†]

[†] In [Men] $\models$ is defined inconsistently with $\vdash$.

3.10 Metatheorem. (The variant [or, dummy renaming] metatheorem) *For any formula $(\forall x)A$, if z does* not *occur in it (i.e., is neither free nor bound), then* $\vdash (\forall x)A \equiv (\forall z)A[x := z]$.

NB. We often write this (under the stated conditions) as $\vdash (\forall x)A[x] \equiv (\forall z)A[z]$.

Proof. Since z is substitutable in x under the stated conditions, $A[x := z]$ is, of course, defined. Thus, by **Ax2** and modus ponens

$$(\forall x)A \vdash A[x := z]$$

By 3.5—since z is not free in $(\forall x)A$—we also have

$$(\forall x)A \vdash (\forall z)A[x := z]$$

By the Deduction Theorem,

$$\vdash (\forall x)A \Rightarrow (\forall z)A[x := z]$$

Noting that x is not free in $(\forall z)A[x := z]$ and is substitutable in z (in $A[x := z]$)—indeed, $A[x := z][z := x] = A$—we can repeat the above argument to get $(\forall z)A[x := z] \vdash A$, hence (by 3.5) $(\forall z)A[x := z] \vdash (\forall x)A$, and, finally, $\vdash (\forall z)A[x := z] \Rightarrow (\forall x)A$. □

Why is $A[x := z][z := x] = A$? We can see this by induction on A (recall that z occurs as neither free nor bound in A).

If A is atomic, the claim is trivial. The claim also clearly "propagates" with the propositional formation rules.

Consider then the case that $A = (\forall w)B$. Note that $w = x$ is possible under our assumptions, but $w = z$ is not. If $w = x$, then $A[x := z] = A$, in particular, z is not free in A, hence $A[x := z][z := x] = A$ as well. So let us work with $w \neq x$. By I.H. $B[x := z][z := x] = B$. Now

$$\begin{aligned}
A[x := z][z := x] &= ((\forall w)B)[x := z][z := x] \\
&= ((\forall w)B[x := z])[z := x] && \langle \text{see 2.5—}w \neq z\rangle \\
&= ((\forall w)B[x := z][z := x]) && \langle \text{see 2.5—}w \neq x\rangle \\
&= ((\forall w)B) && \langle \text{I.H.}\rangle \\
&= A
\end{aligned}$$

We conclude this section with a couple of useful metatheorems.

3.11 Lemma. *If x is not free in A, then* $\vdash A \equiv (\forall x)A$.

Proof. This trivial fact (**Ax2** and **Ax3** and tautological implication) is only stated here to make it "quotable". □

3.12 Metatheorem. *If* $\Gamma \vdash A \Rightarrow B$ *with a condition on* x*, then* $\Gamma \vdash A \Rightarrow (\forall x)B$.

The condition on x *is:* x *is not free in* A*, and there is a* Γ*-proof of* $A \Rightarrow B$*, such that no* Γ*-formula of that proof contains* x *free.*

Proof. Apply 3.5 to conclude $\Gamma \vdash (\forall x)(A \Rightarrow B)$. Thus,

$$\begin{array}{ll} & (\forall x)(A \Rightarrow B) \\ \vdash & \langle \textbf{Ax4} \text{ and modus ponens} \rangle \\ & (\forall x)A \Rightarrow (\forall x)B \\ = & \langle \text{PSL and 3.11} \rangle \\ & A \Rightarrow (\forall x)B \end{array}$$

By EQN, $\Gamma \vdash A \Rightarrow (\forall x)B$. □

By 3.8, *for any two formulas* A and B, $\vdash$ and $\Rightarrow$ are "interchangeable" (strictly speaking, $\vdash A \Rightarrow B$ iff $A \vdash B$).

For this reason, ***assuming that*** $\Rightarrow$ ***is conjunctional when and only when it is used at the left margin of an annotated proof***,† the above proof could be re-written using $\Rightarrow$ (the latter notation seems to be preferred in [GS1, Gr]). The hints have to change though!

$$\begin{array}{ll} & (\forall x)(A \Rightarrow B) \\ \Rightarrow & \langle \textbf{Ax4} \rangle \\ & (\forall x)A \Rightarrow (\forall x)B \\ = & \langle \text{PSL and 3.11} \rangle \\ & A \Rightarrow (\forall x)B \end{array} \tag{1}$$

Note that we dropped the justification "modus ponens".

Of course, in any proof like the following

$$\begin{array}{ll} & A_1 \\ \circ & \langle \text{Hints} \rangle \\ & A_2 \\ \circ & \langle \text{Hints} \rangle \\ & A_3 \\ \circ & \langle \text{Hints} \rangle \\ \vdots & \\ \circ & \langle \text{Hints} \rangle \\ & A_n \end{array}$$

†The "standard" $\Rightarrow$ is, of course, *not* conjunctional: E.g., $p \Rightarrow q \Rightarrow r$ does *not* say $(p \Rightarrow q) \wedge (q \Rightarrow r)$.

where $\circ \in \{\Rightarrow, =\}$—both used conjunctionally—it follows by tautological implication that $\Gamma \vdash A_1 \Rightarrow A_n$, where Γ is the set of (nonlogical) axioms that made the above "calculation" tick.

Therefore, if moreover $\Gamma \vdash A_1$, then (by modus ponens) we conclude that $\Gamma \vdash A_n$.

Taking all this for granted, we normally terminate a calculational proof, such as the one ending with (1) above, without any additional comment (contrast with the proof of 3.12)

3.13 Corollary. *If x is not free in A and $\vdash A \Rightarrow B$, then $\vdash A \Rightarrow (\forall x)B$.*

3.14 Corollary. *If x is not free in A, then $\vdash (\forall x)(A \Rightarrow B) \equiv A \Rightarrow (\forall x)B$.*

Proof. ($\Rightarrow$) This was done in the proof of 3.12.

($\Leftarrow$) We assume then $A \Rightarrow (\forall x)B$, and prove $(\forall x)(A \Rightarrow B)$.

Now, $A \Rightarrow (\forall x)B \vdash A \Rightarrow B$ by $\vdash (\forall x)B \Rightarrow B$ and tautological implication. Since x is not free in $A \Rightarrow (\forall x)B$, we are done by 3.5 and 3.8. $\Box$

3.15 Corollary. *Suppose we have a Γ-proof of $A \Rightarrow B$, where x does not occur free in whatever nonlogical axioms were used. Then $\Gamma \vdash (\forall x)A \Rightarrow (\forall x)B$.*

Proof. By 3.5, $\Gamma \vdash (\forall x)(A \Rightarrow B)$. The result follows by **Ax4** and modus ponens. $\Box$

3.16 Corollary. *The duals of 3.12–3.15 hold.*

Proof. A trivial exercise, using the definition of the "text" $((\exists x)A)$, namely, $(\neg(\forall x)(\neg A))$. $\Box$

4. Derived Leibniz rules

In this section we aim to increase our flexibility in carrying out calculational proofs, by introducing some *derived* rules of the type "Leibniz". Let us start with a version we do *not* have (Strong Leibniz with Uniform Substitution—recall that "strong" means that the premise is an arbitrary formula $A \equiv B$).

$$\frac{A \equiv B}{C[p \setminus A] \equiv C[p \setminus B]} \qquad (SLUS)$$

That SLUS is "invalid" in our Logic follows from 3.4 in [To], "strong generalization", which is a derived rule *if SLUS is available*. But we have seen that E-logic does not support strong generalization.

4.1 Metatheorem. (Strong Leibniz with Contextual Substitution—SLCS) *The following is a derived rule:*

$$\frac{A \equiv B}{C[p := A] \equiv C[p := B]} \tag{$SLCS$}$$

Proof. This was proved in [To] by induction on the formula C.

Basis. C is atomic. If $C = p$, then $C[p := A] = A$ and $C[p := B] = B$, so our conclusion is our hypothesis.

In all other cases $C[p := A] \equiv C[p := B]$ is the tautology $C \equiv C$.

Induction Step(s) (I.S.).

I.S.1. $C = \neg D$. By I.H., $A \equiv B \vdash D[p := A] \equiv D[p := B]$.

Since $D[p := A] \equiv D[p := B] \models_{\mathbf{Taut}} \neg D[p := A] \equiv \neg D[p := B]$, we are done in this case.

I.S.2. $C = D \circ G$ ($\circ \in \{\wedge, \vee, \Rightarrow, \equiv\}$). By I.H., $A \equiv B \vdash D[p := A] \equiv D[p := B]$ and $A \equiv B \vdash G[p := A] \equiv G[p := B]$. Since $D[p := A] \equiv D[p := B], G[p := A] \equiv G[p := B] \models_{\mathbf{Taut}} (D \circ G)[p := A] \equiv (D \circ G)[p := B]$, we are done once more.

I.S.3. $C = (\forall x)D$. By I.H., $A \equiv B \vdash D[p := A] \equiv D[p := B]$, hence (Tautology Theorem) $A \equiv B \vdash D[p := A] \Rightarrow D[p := B]$.

Since x is not free in $A \equiv B$, 3.15 yields $A \equiv B \vdash (\forall x)D[p := A] \Rightarrow (\forall x)D[p := B]$.

Similarly (from $A \equiv B \vdash D[p := A] \Leftarrow D[p := B]$), $A \equiv B \vdash (\forall x)D[p := A] \Leftarrow (\forall x)D[p := B]$, hence, one more application of the Tautology Theorem gives $A \equiv B \vdash (\forall x)D[p := A] \equiv (\forall x)D[p := B]$.

□

4.2 Metatheorem. (Weak Leibniz with Uniform Substitution—WLUS) *The following is a derived rule: If* $\vdash A \equiv B$ *then* $\vdash C[p \setminus A] \equiv C[p \setminus B]$.

Proof. The proof is as above, with the following differences:

We assume $\vdash A \equiv B$ throughout. The I.H. then reads that "$\vdash D[p \setminus A] \equiv D[p \setminus B]$, for all *immediate* subformulas, D, of C".

The induction step **I.S.3** now reads: Let $C = (\forall x)D$. By I.H. we have $\vdash D[p \setminus A] \equiv D[p \setminus B]$. By the Tautology Theorem $\vdash D[p \setminus A] \Rightarrow D[p \setminus B]$, thus $\vdash (\forall x)D[p \setminus A] \Rightarrow (\forall x)D[p \setminus B]$ by 3.15 ($\Gamma = \emptyset$).

Similarly we obtain $\vdash (\forall x)D[p \setminus A] \Leftarrow (\forall x)D[p \setminus B]$ and are done by the Tautology Theorem. □

5. *Monotonicity*

5.1 Definition. We define a set of strings, the *I-Forms* and *D-Forms*, by induction. It is the smallest set of strings over the alphabet $V \cup \{*\}$, where $*$ is *a new symbol added to the alphabet* V, satisfying the following:

Form1. (*Basis*) $*$ is an I-Form

Form2. If A is any formula and $\mathcal{U}$ is an I-Form (respectively, D-Form), then the following are also I-Forms (respectively, D-Forms): $(\mathcal{U} \vee A)$, $(A \vee \mathcal{U})$, $(\mathcal{U} \wedge A)$, $(A \wedge \mathcal{U})$, $(A \Rightarrow \mathcal{U})$, $((\forall x)\mathcal{U})$ and $((\exists x)\mathcal{U})$, while the following are D-Forms (respectively, I-Forms): $(\neg\mathcal{U})$ and $(\mathcal{U} \Rightarrow A)$.

We will just say $\mathcal{U}$ is a Form, if we do not wish to spell out *its type* (I or D). We will use calligraphic capital letters $\mathcal{U}, \mathcal{V}, \mathcal{W}, \mathcal{X}, \mathcal{Y}$ to denote Forms. □

5.2 Definition. For any Form $\mathcal{U}$ and any formula A or form $\mathcal{W}$, the symbols $\mathcal{U}[A]$ and $\mathcal{U}[\mathcal{W}]$ mean, respectively, the result of the uniform substitutions $\mathcal{U}[* \setminus A]$ and $\mathcal{U}[* \setminus \mathcal{W}]$. □

Our I-Forms and D-Forms—"I" for increasing, and "D" for decreasing—are motivated by, but are different from,† the Positive and Negative Forms of Schütte [Schü].

The expected behaviour of the Forms is that they are "monotonic functions" of the $*$-"variable" in the following sense: We expect that $\vdash A \Rightarrow B$ will imply $\vdash \mathcal{U}[A] \Rightarrow \mathcal{U}[B]$ if $\mathcal{U}$ is an I-Form, and $\vdash \mathcal{U}[A] \Leftarrow \mathcal{U}[B]$ if it is a D-Form.

Now, $\Rightarrow$ is "like" $\leq$ in Boolean algebras, the latter defined by "$a \leq b$ means $a \vee b = b$" (compare with [GS1], Axiom 3.57 for $\Rightarrow$). This observation justifies the terminology "monotonic functions".

We now pursue in detail the intentions stated in the above remark.

5.3 Lemma. *Every Form contains exactly one occurrence of* $*$ *as a substring.*

Proof. Induction on Forms. The basis is immediate. Moreover, the property we are asked to prove obviously "propagates" with the formation rules. □

5.4 Lemma. *For any Form* $\mathcal{U}$ *and formula* A*,* $\mathcal{U}[A]$ *is a formula.*

Proof. Induction on Forms. The basis is obvious, and clearly the property propagates with the formation rules. □

---

†For example, $(* \wedge A)$ is an I-Form but not a Positive Form in the sense of [Schü], since the latter would necessitate, in particular, that $(\mathit{true} \wedge A)$ be a tautology.

5.5 Lemma. *No Form has both types I and D.*

Proof. Induction on Forms (really using the least principle and proof by contradiction). Let $\mathcal{U}$ have the least complexity among forms that have both types. This is not the “basic” Form $*$ as that is declared to have just type I.

Can it be a form $(\mathcal{V} \vee A)$? No, for it has both types I and D, so that also $\mathcal{V}$ must have both types I and D, contradicting the assumption that $\mathcal{U}$ was the least complex schizophrenic Form. We obtain similar contradictions in the case of all the other formation rules. $\square$

5.6 Lemma. *For any Forms $\mathcal{U}$ and $\mathcal{V}$, we have the following* composition *properties:*

(1) *If $\mathcal{U}$ is an I-Form, then $\mathcal{U}[\mathcal{V}]$ has the type of $\mathcal{V}$.*

(2) *If $\mathcal{U}$ is an D-Form, then $\mathcal{U}[\mathcal{V}]$ has the type* opposite *to that of $\mathcal{V}$.*

Proof. We do induction on $\mathcal{U}$ to prove (1) and (2) simultaneously. The basis is obvious, as $\mathcal{U} = *$, hence $\mathcal{U}[\mathcal{V}] = \mathcal{V}$.

Case 1. $\mathcal{U} = (\mathcal{W} \vee A)$, for some $A \in \mathbf{Wff}$. $\mathcal{U}[\mathcal{V}] = (\mathcal{W}[\mathcal{V}] \vee A)$. $\mathcal{U}$ and $\mathcal{W}$ have the same type.

By I.H. and the definition of Forms, the claim follows.

We omit a few similar cases ...

Case 2. $\mathcal{U} = (\mathcal{W} \Rightarrow A)$, for some $A \in \mathbf{Wff}$. $\mathcal{U}[\mathcal{V}] = (\mathcal{W}[\mathcal{V}] \Rightarrow A)$. $\mathcal{U}$ and $\mathcal{W}$ have opposite types.

By I.H. and the definition of Forms, the claim follows.

We omit a few similar cases ...

Case 3. $\mathcal{U} = ((\forall x)\mathcal{W})$. $\mathcal{U}[\mathcal{V}] = ((\forall x)\mathcal{W}[\mathcal{V}])$. $\mathcal{U}$ and $\mathcal{W}$ have the same type.

By I.H. and the definition of Forms, the claim follows.

$\square$

5.7 Remark. Thus, if $\mathcal{U}$ is obtained by a chain of compositions, it is an I-Form if the chain contains an even number of D-Forms, it will be a D-Form otherwise.

For example, if $\mathcal{U}$ is a D-Form, then $\mathcal{U}[A \Rightarrow *]$ is still a D-Form, but $\mathcal{U}[* \Rightarrow A]$ is an I-Form.

5.8 Metatheorem. (Monotonicity and Antimonotonicity) *Let* $\vdash A \Rightarrow B$.

If $\mathcal{U}$ is an I-Form, then $\vdash \mathcal{U}[A] \Rightarrow \mathcal{U}[B]$, else (a D-Form) $\vdash \mathcal{U}[A] \Leftarrow \mathcal{U}[B]$.

We call MON the rule "if $\vdash A \Rightarrow B$ and $\mathcal{U}$ is an I-Form, then $\vdash \mathcal{U}[A] \Rightarrow \mathcal{U}[B]$".

We call AMON the rule "if $\vdash A \Rightarrow B$ and $\mathcal{U}$ is an D-Form, then $\vdash \mathcal{U}[A] \Leftarrow \mathcal{U}[B]$".

Proof. Induction on $\mathcal{U}$.

Basis. $\mathcal{U} = *$, hence we want to prove $\vdash A \Rightarrow B$, which is the same as the hypothesis.

The induction steps:

Case 1. $\mathcal{U} = (\mathcal{W} \vee C)$, for some $C \in$ **Wff**. If $\mathcal{W}$ is an I-Form, then (I.H.) $\vdash \mathcal{W}[A] \Rightarrow \mathcal{W}[B]$, hence $\vdash (\mathcal{W}[A] \vee C) \Rightarrow (\mathcal{W}[B] \vee C)$ by tautological implication.

If $\mathcal{W}$ is a D-Form, then (I.H.) $\vdash \mathcal{W}[A] \Leftarrow \mathcal{W}[B]$, hence $\vdash (\mathcal{W}[A] \vee C) \Leftarrow (\mathcal{W}[B] \vee C)$ by tautological implication.

Case 2. $\mathcal{U} = (\mathcal{W} \Rightarrow C)$, for some $C \in$ **Wff**. If $\mathcal{W}$ is an I-Form, then (I.H.) $\vdash \mathcal{W}[A] \Rightarrow \mathcal{W}[B]$, hence $\vdash (\mathcal{W}[A] \Rightarrow C) \Leftarrow (\mathcal{W}[B] \Rightarrow C)$ by tautological implication.

If $\mathcal{W}$ is a D-Form, then (I.H.) $\vdash \mathcal{W}[A] \Leftarrow \mathcal{W}[B]$, hence $\vdash (\mathcal{W}[A] \Rightarrow C) \Rightarrow (\mathcal{W}[B] \Rightarrow C)$ by tautological implication.

We omit a few similar cases based on tautological implication ...

Case 3. $\mathcal{U} = ((\forall x)\mathcal{W})$. If $\mathcal{W}$ is an I-Form, then (I.H.) $\vdash \mathcal{W}[A] \Rightarrow \mathcal{W}[B]$. By 3.15, $\vdash ((\forall x)\mathcal{W}[A]) \Rightarrow ((\forall x)\mathcal{W}[B])$.

If $\mathcal{W}$ is a D-Form, then (I.H.) $\vdash \mathcal{W}[A] \Leftarrow \mathcal{W}[B]$.

By 3.15, $\vdash ((\forall x)\mathcal{W}[A]) \Leftarrow ((\forall x)\mathcal{W}[B])$.

Case 4. $\mathcal{U} = ((\exists x)\mathcal{W})$. As above, but relying on 3.16 instead.

□

MON and AMON are applied *after* we have eliminated the presence of $\equiv$ from formulas.

5.9 Example. We illustrate the use of the rules MON or AMON by revisiting the calculational proof fragment of 3.12.

$$
\begin{array}{ll}
 & (\forall x)(A \Rightarrow B) \\
\Rightarrow & \langle \textbf{Ax4} \rangle \\
 & (\forall x)A \Rightarrow (\forall x)B \\
\Rightarrow & \langle \text{AMON—on } * \Rightarrow (\forall x)B\text{—and } \textbf{Ax3} \rangle \\
 & A \Rightarrow (\forall x)B
\end{array}
$$

□

If we are willing to weaken the type of substitution we effect into Forms, we can strengthen the type of premise in MON and AMON:

5.10 Metatheorem. ("Strong" MON/AMON with contextual substitution) *The following are derived rules:*

$$A \Rightarrow B \vdash \mathcal{U}[* := A] \Rightarrow \mathcal{U}[* := B],\ \textit{if}\ \mathcal{U}\ \textit{is an I-Form}$$

$$A \Rightarrow B \vdash \mathcal{U}[* := A] \Leftarrow \mathcal{U}[* := B],\ \textit{if}\ \mathcal{U}\ \textit{is a D-Form}$$

Proof. The proof is as in 5.8, except that the induction steps under cases 3 and 4 are modified as follows:

Case 3. $\mathcal{U} = ((\forall x)\mathcal{W})$. If $\mathcal{W}$ is an I-Form, then (I.H.)

$$A \Rightarrow B \vdash \mathcal{W}[* := A] \Rightarrow \mathcal{W}[* := B] \qquad (i)$$

We want to argue that

$$A \Rightarrow B \vdash (\forall x)\mathcal{W}[* := A] \Rightarrow (\forall x)W[* := B] \qquad (ii)$$

where we have already incorporated $((\forall x)\mathcal{W})[* := A] = (\forall x)(\mathcal{W}[* := A])$, etc., and then dropped the unnecessary brackets.

Now, if the substitutions in (ii) are not defined, then there is nothing to state (let alone prove).

Assuming that they *are* defined, then $A \Rightarrow B$ has no free occurrence of x. By 3.15, (ii) follows from (i).

If $\mathcal{W}$ is a D-Form, then we argue as above on the I.H.

$$A \Rightarrow B \vdash \mathcal{W}[* := A] \Leftarrow \mathcal{W}[* := B].$$

Case 4. $\mathcal{U} = ((\exists x)\mathcal{W})$. We use here 3.16 instead of 3.15.

□

The above is as far as it goes. If we allow uniform substitution as well, to obtain "Extra strong MON/AMON", then the rule yields strong generalization, hence it is not valid in E-logic. The following calculation illustrates this point.

$$\begin{array}{cl}
 & A \\
= & \langle A \equiv \mathit{true} \Rightarrow A \text{ is a tautology}\rangle \\
 & \mathit{true} \Rightarrow A \\
\Rightarrow & \langle \text{“Extra strong MON” and the I-Form } (\forall x)* \rangle \\
 & (\forall x)\mathit{true} \Rightarrow (\forall x)A \\
= & \langle \text{PSL and 3.11} \rangle \\
 & \mathit{true} \Rightarrow (\forall x)A \\
= & \langle A \equiv \mathit{true} \Rightarrow A \text{ is a tautology}\rangle \\
 & (\forall x)A
\end{array}$$

The above $\Rightarrow$ (on the left margin) is, of course, $\vdash$ (see the -passage following 3.12). Thus we have just "proved" $A \vdash (\forall x)A$.

6. *Soundness and Completeness of E-logic*

We introduce semantics that accurately reflect our syntactic choices (most importantly, to disallow strong generalization). In particular, we will define "logically implies", $\models$, so that $\Gamma \vdash A$ iff $\Gamma \models A$.[†] Since we have an unconstrained Deduction Theorem that says "$A \vdash B$ iff $\vdash A \Rightarrow B$", we need our semantics to also say "$A \models B$ iff $\models A \Rightarrow B$".

Thus the semantics defined here will be *different from those in* [To].

We still want to keep the "English" definition of $A \models B$ identical to that in [To]: "Every model of A is a model of B". For that reason the term *model* will mean here something else! (See [Schü]; the two definitions of model are identical if we only deal with sentences).

6.1 Definition. Given a language $L = (V, \mathbf{Term}, \mathbf{Wff})$, a *structure* $\mathfrak{M} = (M, \mathcal{I})$ *appropriate for* L is such that $M \neq \emptyset$ is a set (the "domain") and $\mathcal{I}$ is a mapping that assigns

(1) to each object variable x of V a unique member $x^{\mathcal{I}} \in M$

(2) to each constant a of V a unique member $a^{\mathcal{I}} \in M$

(3) to each function f of V—of arity n—a unique (total) function $f^{\mathcal{I}} : M^n \to M$

(4) to each predicate P of V—of arity n—a unique set $P^{\mathcal{I}} \subseteq M^n$

(5) to each propositional variable p of V a unique member $p^{\mathcal{I}}$ of the two element set $\{\mathbf{t}, \mathbf{f}\}$ (we understand $\mathbf{t}$ as "true" and $\mathbf{f}$ as "false")

(6) moreover we set $\mathit{true}^{\mathcal{I}} = \mathbf{t}$ and $\mathit{false}^{\mathcal{I}} = \mathbf{f}$, where the use of "=" here is metamathematical (equality on $\{\mathbf{t}, \mathbf{f}\}$).

□

Item (1) makes the difference between the definition of semantics here and in [To].

6.2 Definition. Given L and a structure $\mathfrak{M} = (M, \mathcal{I})$ appropriate for L. $L(\mathfrak{M})$ denotes the language obtained from L by *adding* in V a unique *name* $\hat{i}$ for each object $i \in M$. This amends both sets $\mathbf{Term}$, $\mathbf{Wff}$ into $\mathbf{Term}(\mathfrak{M})$, $\mathbf{Wff}(\mathfrak{M})$. Members of the latter sets are called $\mathfrak{M}$-terms and $\mathfrak{M}$-formulas respectively.

We extend $\mathcal{I}$ to the new constants: $\hat{i}^{\mathcal{I}} = i$ for all $i \in M$ (where the metamathematical "=" is that on M). □

All we have done here is to allow ourselves to do substitutions like $[x := i]$ formally. We do instead, $[x := \hat{i}]$. One next gives "meaning" to *all* terms in $L(\mathfrak{M})$. We do not restrict this to just *closed* terms (as it was done in [To]) since here we are "freezing" object variables anyhow (we instantiate them via $\mathcal{I}$).

[†]We will not verify this "strong" Gödel completeness that is based on "compactness".

6.3 Definition. For terms t in $\mathbf{Term}(\mathfrak{M})$ we define the symbol $t^{\mathcal{I}} \in M$ inductively:

(1) If t is any of x (object variable), a (original constant), or $\hat{i}$ (imported constant), then $t^{\mathcal{I}}$ has already been defined.

(2) If t is the string $ft_1 \ldots t_n$, where f is n-ary, and $t_1, \ldots, t_n$ are $\mathfrak{M}$-terms, we define $t^{\mathcal{I}}$ to be the object (of M) $f^{\mathcal{I}}(t_1^{\mathcal{I}}, \ldots, t_n^{\mathcal{I}})$.

□

Finally, we give meaning to *all* $\mathfrak{M}$*-formulas*, again not restricting attention to just sentences.

6.4 Definition. For *any* formula A in $\mathbf{Wff}(\mathfrak{M})$ we define the symbol $A^{\mathcal{I}}$ inductively. In all cases, $A^{\mathcal{I}} \in \{\mathbf{t}, \mathbf{f}\}$.

(1) If A is any of p or *true* or *false*, then $A^{\mathcal{I}}$ has already been defined.

(2) If A is the string $t \approx s$, where t and s are $\mathfrak{M}$-terms, then $A^{\mathcal{I}} = \mathbf{t}$ iff $t^{\mathcal{I}} = s^{\mathcal{I}}$ (again, the last two occurrences of $=$ refer to equality on $\{\mathbf{t}, \mathbf{f}\}$ and M respectively).

(3) If A is the string $Pt_1 \ldots t_n$, where P is an n-ary predicate and the t_i are $\mathfrak{M}$-terms, then $A^{\mathcal{I}} = \mathbf{t}$ iff $(t_1^{\mathcal{I}}, \ldots, t_n^{\mathcal{I}}) \in P^{\mathcal{I}}$.

(4) If A is any of $\neg B, B \wedge C, B \vee C, B \Rightarrow C, B \equiv C$, then $A^{\mathcal{I}}$ is determined by the usual truth tables using the values $B^{\mathcal{I}}$ and $C^{\mathcal{I}}$.

(5) If A is $(\forall x)B$, then $A^{\mathcal{I}} = \mathbf{t}$ iff $(B[x := \hat{i}])^{\mathcal{I}} = \mathbf{t}$ *for all* $i \in M$.

Of course, the above also gives meaning to $t^{\mathcal{I}}$ and $A^{\mathcal{I}}$ for any terms and formulas over the original language, since $\mathbf{Term}(\mathfrak{M}) \supseteq \mathbf{Term}$ and $\mathbf{Wff}(\mathfrak{M}) \supseteq \mathbf{Wff}$. □

We have "imported" constants from M into L in order to be able to state the semantics of $(\forall x)B$ above in the simple manner we just did (following [Schü]).

6.5 Definition. Let $A \in \mathbf{Wff}$ and $\mathfrak{M}$ be a structure as above.

We say that A is *satisfiable in* $\mathfrak{M}$, or that $\mathfrak{M}$ is a *model of* A[†]—in symbols $\models_{\mathfrak{M}} A$—iff $A^{\mathcal{I}} = \mathbf{t}$.

For any set of formulas Γ from $\mathbf{Wff}$, $\models_{\mathfrak{M}} \Gamma$ denotes the sentence "$\mathfrak{M}$ *is a model of* Γ", and means that *for all* $A \in \Gamma$, $\models_{\mathfrak{M}} A$.

A formula A is *universally valid* (we often say just *valid*) iff *every structure appropriate for the language is a model of* A. *In particular, that says that fixing* M, A *is "true in all states"* $\mathcal{I}$.

Under these circumstances we simply write $\models A$. □

[†]Contrast with the "models" in [To].

6.6 Definition. We say that Γ *logically implies* A, in symbols $\Gamma \models A$, to mean that *every model of* Γ *is also a model of* A. $\Box$

Clearly, in the case of $A \models B$ the above says that, having fixed the domain M, every "state" $\mathcal{I}$ that makes A true makes B true. Thus, $A \models B$ exactly when $\models A \Rightarrow B$, as earlier promised.

6.7 Definition. (*First order theories*) A *(first order) theory* is a collection of the following objects:

Theory1. A first order language $L = (V, \mathbf{Term}, \mathbf{Wff})$,

Theory2. A set of logical axioms,

Theory3. A set of rules of inference,

Theory4. A set of nonlogical axioms, plus a definition of "deduction" (proof) and "theorem" (2.12, and 2.10).

We often *name* the theory by the name of its nonlogical axioms (as in "let Γ be a theory ...", in which case we write $\Gamma \vdash A$ to indicate that A is a Γ-theorem), but we may also name the theory by other characteristics, e.g., the choice of language. For example, we may have two theories under discussion, that only differ in the choice of the language (L vs., say, L'). We may call one the theory T and the other the theory T', in which case we indicate where deductions take place by writing $\Gamma \vdash_T A$ or $\Gamma \vdash_{T'} A$ as the case may be. Similarly, all other things might be the same, except that the choice of rules of inference (or logical axioms) is different. Again, we choose names to reflect these different choices. We have already used such notation and terminology: E-logic and En-logic. We now are saying that we may also use the terminology "E-theory" and "En-theory".

A *pure theory* is one with $\Gamma = \emptyset$. $\Box$

6.8 Remark. Remarks embedded in the above definition justify the use of the indefinite article in "*A* pure theory ...".

6.9 Definition. (*Soundness*) A *pure* theory is *sound*, iff $\vdash A$ implies $\models A$, that is, iff all the theorems of the theory are universally valid. $\Box$

Towards the soundness result below we carefully look at two nastily tedious (but easy) lemmata.

6.10 Lemma. *Given a term t, variables $x \neq y$, where y is* not in t, *and a constant a. Then, for any term s and formula A, $s[x := t][y := a] = s[y := a][x := t]$ and $A[x := t][y := a] = A[y := a][x := t]$.*

Proof. (*Induction on s*):
Basis.

$$s[x := t][y := a] = \begin{cases} s = x & \text{then } t \\ s = y & \text{then } a \\ s = z & \text{where } z \notin \{x, y\}, \text{ then } z \\ s = b & \text{then } b \end{cases}$$

$$= s[y := a][x := t]$$

For the induction step let $s = fr_1 \ldots r_n$, where f has arity n. Then

$$\begin{aligned} s[x := t][y := a] &= fr_1[x := t][y := a] \ldots r_n[x := t][y := a] \\ &= fr_1[y := a][x := t] \ldots r_n[y := a][x := t] \qquad \text{by I.H.} \\ &= s[y := a][x := t] \end{aligned}$$

Induction on A):
Basis.

$$A[x := t][y := a] = \begin{cases} A = p & \text{then } p \\ A = \mathit{true} & \text{then } \mathit{true} \\ A = \mathit{false} & \text{then } \mathit{false} \\ A = Pr_1 \ldots r_n & \text{then} \\ & Pr_1[x := t][y := a] \ldots r_n[x := t][y := a] = \\ & \quad Pr_1[y := a][x := t] \ldots r_n[y := a][x := t] \\ A = r \approx s & \text{then } r[x := t][y := a] \approx s[x := t][y := a] \\ & \quad = r[y := a][x := t] \approx s[y := a][x := t] \end{cases}$$

$$= A[y := a][x := t]$$

The property we are proving, trivially, propagates with boolean connectives. Let us do the induction step just in the case where $A = (\forall w)B$. If $w = x$ the result is trivial. Otherwise,

$$\begin{aligned} A[x := t][y := a] &= ((\forall w)B)[x := t][y := a] \\ &= ((\forall w)B[x := t][y := a]) \\ &= ((\forall w)B[y := a][x := t]) \text{ by I.H.} \\ &= ((\forall w)B)[y := a][x := t] \\ &= A[y := a][x := t] \end{aligned}$$

□

6.11 Lemma. *Given a structure $\mathfrak{M} = (M, \mathcal{I})$, a term s, and formula A over $L(\mathfrak{M})$.*

Let t be another term over $L(\mathfrak{M})$, such that $t^{\mathcal{I}} = i \in M$.

Then, $(s[x := t])^{\mathcal{I}} = (s[x := \hat{\imath}])^{\mathcal{I}}$ and $(A[x := t])^{\mathcal{I}} = (A[x := \hat{\imath}])^{\mathcal{I}}$, in the latter case on the assumption that $A[x := t]$ is defined.

This *almost* says the intuitively expected (but formally incorrect): $(A[t])^{\mathcal{I}} = A^{\mathcal{I}}[t^{\mathcal{I}}]$.

Proof. (*Induction on s*):

Basis. $s[x := t] = s$ if $s \in \{y, a, \hat{\jmath}\}$ $(y \neq x)$. Hence $(s[x := t])^{\mathcal{I}} = s^{\mathcal{I}} = (s[x := \hat{\imath}])^{\mathcal{I}}$ in this case. If $s = x$, then $x[x := t] = t$ and $x[x := \hat{\imath}] = \hat{\imath}$, and the claim follows once more.

For the induction step let $s = fr_1 \dots r_n$, where f has arity n. Then

$$\begin{aligned}
(s[x := t])^{\mathcal{I}} &= f^{\mathcal{I}}\big((r_1[x := t])^{\mathcal{I}}, \dots, (r_n[x := t])^{\mathcal{I}}\big) \\
&= f^{\mathcal{I}}\big((r_1[x := \hat{\imath}])^{\mathcal{I}}, \dots, (r_n[x := \hat{\imath}])^{\mathcal{I}}\big) && \text{by I.H.} \\
&= (s[x := \hat{\imath}])^{\mathcal{I}}
\end{aligned}$$

(*Induction on A*):

Basis.

$$A[x := t] = \begin{cases} A = p & \text{then } p \\ A = \mathit{true} & \text{then } \mathit{true} \\ A = \mathit{false} & \text{then } \mathit{false} \end{cases}$$

$$= A = A[x := \hat{\imath}]$$

Thus the claim follows in the above cases.

If $A = Pr_1 \dots r_n$, then†

$$\begin{aligned}
(A[x := t])^{\mathcal{I}} &= P^{\mathcal{I}}\big((r_1[x := t])^{\mathcal{I}}, \dots, (r_n[x := t])^{\mathcal{I}}\big) \\
&= P^{\mathcal{I}}\big((r_1[x := \hat{\imath}])^{\mathcal{I}}, \dots, (r_n[x := \hat{\imath}])^{\mathcal{I}}\big) \\
&= (A[x := \hat{\imath}])^{\mathcal{I}}
\end{aligned}$$

Similarly if $A = r \approx s$.

The property we are proving, clearly, propagates with boolean connectives. Let us do the induction step just in the case where $A = (\forall w)B$. If $w = x$ the result is trivial. Otherwise, we note that—since we assume that t is substitutable

†For a metamathematical relation Q, as usual, $Q(a, b, \dots) = \mathbf{t}$ stands for $(a, b, \dots) \in Q$.

in x—w does not occur in t, and proceed as follows:

$$
\begin{aligned}
(A[x := t])^{\mathcal{I}} = \mathbf{t} \text{ iff } & \big(((\forall w)B)[x := t]\big)^{\mathcal{I}} = \mathbf{t} \\
\text{iff } & \big(((\forall w)B[x := t])\big)^{\mathcal{I}} = \mathbf{t} \\
\text{iff } & (B[x := t][w := \hat{j}])^{\mathcal{I}} = \mathbf{t} \text{ for all } j \in M, \text{ by 6.4(5)} \\
\text{iff } & (B[w := \hat{j}][x := t])^{\mathcal{I}} = \mathbf{t} \text{ for all } j \in M, \text{ by 6.10} \\
\text{iff } & \big((B[w := \hat{j}])[x := t]\big)^{\mathcal{I}} = \mathbf{t} \text{ for all } j \in M \\
\text{iff } & \big((B[w := \hat{j}])[x := \hat{i}]\big)^{\mathcal{I}} = \mathbf{t} \text{ for all } j \in M, \text{ by I.H.} \\
\text{iff } & (B[w := \hat{j}][x := \hat{i}])^{\mathcal{I}} = \mathbf{t} \text{ for all } j \in M \\
\text{iff } & (B[x := \hat{i}][w := \hat{j}])^{\mathcal{I}} = \mathbf{t} \text{ for all } j \in M, \text{ by 6.10} \\
\text{iff } & \big(((\forall w)B[x := \hat{i}])\big)^{\mathcal{I}} = \mathbf{t} \text{ by 6.4(5)} \\
\text{iff } & \big(((\forall w)B)[x := \hat{i}]\big)^{\mathcal{I}} = \mathbf{t} \\
\text{iff } & (A[x := \hat{i}])^{\mathcal{I}} = \mathbf{t}
\end{aligned}
$$

□

6.12 Metatheorem. *Any pure E-theory is sound.*

Proof. The pure E-theory over a fixed alphabet L is equivalent to the En-theory over the same alphabet (3.4). Thus the proof proceeds for an En-theory.

Let $\vdash A$. Pick an arbitrary structure $\mathfrak{M} = (M, \mathcal{I})$ appropriate for L and do induction on $\emptyset$-theorems to show that $\models_{\mathfrak{M}} A$.

Basis. A is a logical axiom (see 2.6).

Now, axioms in group **Ax1** are tautologies over prime formulas. That is, regardless of the values $P^{\mathcal{I}}$ of prime formulas P in B, if B is a tautology, then $B^{\mathcal{I}} = \mathbf{t}$ (see 2.2 and 6.4(4)). By 6.4(5), any (partial) generalization, A, of B will also come out $\mathbf{t}$ under $\mathcal{I}$. Thus, $\models_{\mathfrak{M}} A$ in this case.

We next show that if A is a partial generalization of $(\forall x)B \Rightarrow B[x := t]$, then $A^{\mathcal{I}} = \mathbf{t}$, from which follows that $\models_{\mathfrak{M}} A$. We ask the reader to verify the satisfiability—in the arbitrary $\mathfrak{M}$—of all the remaining axioms.

By 6.4(5), it suffices to prove that

$$\big((\forall x)B \Rightarrow B[x := t]\big)^{\mathcal{I}} = \mathbf{t} \tag{1}$$

Arguing by contradiction, let

$$\big((\forall x)B\big)^{\mathcal{I}} = \mathbf{t} \tag{2}$$

but

$$(B[x := t])^{\mathcal{I}} = \mathbf{f} \tag{3}$$

By 6.4(5) and (2), $(B[x := \hat{i}])^{\mathcal{I}} = \mathbf{t}$ for all $i \in M$.

By 6.11 and (3), $(B[x := \hat{i}])^{\mathcal{I}} = \mathbf{f}$, for some $i \in M$, contradicting what we just said one line ago. This proves (1).

Induction step. W show that if $\models C$ and $\models C \Rightarrow A$, then $\models A$.

Indeed, fix an $\mathfrak{M} = (M, \mathcal{I})$ and show that $A^{\mathcal{I}} = \mathbf{t}$. By assumption, $C^{\mathcal{I}} = \mathbf{t}$ and $C^{\mathcal{I}} \Rightarrow A^{\mathcal{I}} = \mathbf{t}$. Modus ponens and truth tables do the rest. □

A by-product of soundness is consistency. An (E-)theory Γ is *consistent* iff $\mathbf{Thm}_\Gamma \subset \mathbf{Wff}$ (proper subset).

Thus, any pure E-theory is consistent, since, by soundness, *false* is not provable.

6.13 Definition. A theory Γ is *complete* iff $\Gamma \models A$ implies $\Gamma \vdash A$ for any formula A. □

We show the completeness of a pure E-theory by proving that it *extends conservatively*† the E-theory obtained by leaving all else the same but dropping all propositional variables and constants from the alphabet V. (The latter is known to be complete [En].)

We employ two technical lemmata:

6.14 Lemma. (Substitution into propositional variables) *In any E-theory (En-theory) Γ, if $\Gamma \vdash A$, with* a condition on the proof*, then $\Gamma \vdash A[p := W]$ for any formula W and propositional variable p.*

The condition is: *The propositional variable p does not occur in any formula of Γ used in the proof of A.*

Proof. Induction on the length n of the Γ-proof of A.

Say that a proof (satisfying the condition) is

$$B_1, \ldots, B_n \tag{1}$$

where $B_n = A$.

Basis. $n = 1$. Suppose that A is a logical axiom. Then $A[p := W]$ is as well (by 2.6), thus $\Gamma \vdash A[p := W]$.

Suppose that A is a nonlogical axiom. Then $A[p := W] = A$ by the condition on the proof, thus $\Gamma \vdash A[p := W]$.

We may assume that we are working in En-logic. On the induction hypothesis that the claim is fine for proof-lengths $< n$, let us address the case of n:

If A (i.e., B_n) is logical or nonlogical, then we have nothing to add.

†A theory T' over the language L' is a conservative extension of a theory T over the language L, if, first of all, every theorem of T is a theorem of T', and (the conservative part) moreover, any theorem of T' *that is over L—the language of T*—is also a theorem of T. That is, T' proves no new theorems in the old language.

So let $B_j = (B_i \Rightarrow A)$ in (1) above, where i and j are each less than n (i.e., the last step of the proof was an application of modus ponens).

By I.H., $\Gamma \vdash B_i[p := W]$ and $\Gamma \vdash B_i[p := W] \Rightarrow A[p := W]$. Thus, by modus ponens, $\Gamma \vdash A[p := W]$. $\square$

6.15 Main Lemma. ([To]) *Let A be a formula over the language L of section 1, and let p be a propositional variable that occurs in A.*

Extend the language L by adding P, a new 1-ary predicate symbol.

Then, $\models A$ *iff* $\models A[p := (\forall x)Px]$ *and* $\vdash A$ *iff* $\vdash A[p := (\forall x)Px]$.

Proof. ($\models$) The only-if is by soundness (substitution 6.14 was used).

For the if-part pick any structure $\mathfrak{M} = (M, \mathcal{I})$ and prove that $A^{\mathcal{I}} = \mathbf{t}$. To this end, expand $\mathfrak{M}$ to $\mathfrak{M}' = (M, \mathcal{I}')$ where $\mathcal{I}'$ is the same as $\mathcal{I}$, *except* that it also gives meaning to P as follows: If $p^{\mathcal{I}} = \mathbf{t}$, set $P^{\mathcal{I}'} = M$, else set $P^{\mathcal{I}'} = \emptyset$. Clearly, $p^{\mathcal{I}'} = ((\forall x)Px)^{\mathcal{I}'}$.

By assumption, $\big(A[p := (\forall x)Px]\big)^{\mathcal{I}'} = \mathbf{t}$, hence, $A^{\mathcal{I}} = A^{\mathcal{I}'} = \mathbf{t}$ as well.

($\vdash$) The only-if is the result of 6.14.

For the if part, let $\vdash A[p := (\forall x)Px]$. By induction on $\emptyset$-theorems we show that $\vdash A$ as well.

Basis. $A[p := (\forall x)Px]$ is a logical axiom. If it is a partial generalization of a formula $B[p := (\forall x)Px]$ in group **Ax1**, then $B[p := (\forall x)Px]$ is a tautology. But then so is B, hence (by 3.6) $\vdash A$.

Ax5 is not applicable (it cannot contain $(\forall x)Px$).

It is clear that if $A[p := (\forall x)Px]$ is a partial generalization of a formula in any of the groups **Ax2**–**Ax4** or **Ax6**, then replacing all occurrences of $(\forall x)Px$ in $A[p := (\forall x)Px]$ by the propositional variable p results to a formula of the same form, so A is still an axiom, hence $\vdash A$.

Modus Ponens. Let $\vdash B[p := (\forall x)Px]$ and $\vdash B[p := (\forall x)Px] \Rightarrow A[p := (\forall x)Px]$. By I.H., $\vdash B$ and $\vdash B \Rightarrow A$.

Hence $\vdash A$. $\square$

6.16 Corollary. ([To]) *Any pure E-theory is complete.*

Proof. Fix attention to the pure E-theory over a fixed language L. Let A be a formula in the language, and let $\models A$.

Denote by A' the formula obtained from A by replacing each occurrence of *true* (respectively *false*) by $p \vee \neg p$ (respectively $p \wedge \neg p$) where p is a propositional variable *not* occurring in A. Let A'' be obtained from A' by replacing each propositional variable $p, q, \ldots$ in it by $(\forall x)Px, (\forall x)Qx, \ldots$ respectively, where $P, Q, \ldots$ are *new* predicate symbols (so we *expand* L by these additions).

Clearly, by 6.15, $\models A''$. This formula is in the language of [En] (which is the same as L of section 1, but it has *no* propositional variables or constants). Thus, by completeness of En-logic/E-logic over such a “restricted” language (proved in [En]), $\vdash A''$, *the proof being carried out in the restricted language.*

But, trivially, this proof is valid over the language L (same axioms, same rules), hence also $\vdash A'$, by 6.15.

Finally, by SLCS—since $\vdash p \vee \neg p \equiv \mathit{true}$ and $\vdash p \wedge \neg p \equiv \mathit{false}$—and EQN, we get $\vdash A$. $\square$

7. Appendix

The reader is referred to [To] where all the axioms in [GS1], chapters 8 and 9, were shown to be *derived* in the logic of [To].

Practically identical proofs are available within our E-logic, and they will not be repeated here.

The justification of uses of generalization will have to be more careful in E-logic (in [To] we could just go ahead with strong generalization). The reader should be able to provide the right wording in each case, "translating" the proofs in [To] to the present setting.

We recall that axiom schemata **Ax5** and **Ax6** are used in the proof of the "one-point rule" (see [To]).

We will only revisit here axiom (9.5) of [GS1]—which is not an axiom of our E-logic—and the Leibniz rules (8.12) of [GS1]. Nomenclature and numbers given in brackets are those in [GS1].

A.1 "Distributivity of $\vee$ over $\forall$ (9.5)". This says that

$$\vdash (\forall x)(A \vee B) \equiv A \vee (\forall x)B \tag{$\vee\forall$}$$

provided that x is not free in A. A proof of ($\vee\forall$) in E-logic follows:

$$\begin{array}{ll} & (\forall x)(A \vee B) \\ = & \langle \text{WLUS and } \vdash A \vee B \equiv \neg A \Rightarrow B \rangle \\ & (\forall x)(\neg A \Rightarrow B) \\ = & \langle 3.14 \rangle \\ & \neg A \Rightarrow (\forall x)B \\ = & \langle \vdash \neg A \Rightarrow (\forall x)B \equiv A \vee (\forall x)B \rangle \\ & A \vee (\forall x)B \end{array}$$

A.2 The twin rules "Leibniz (8.12)" ([GS1], p.148), are stated below. The ones immediately below are the "no-capture" versions, using contextual substitution.

$$\frac{A \equiv B}{(\forall x)(C[p := A] \Rightarrow D) \equiv (\forall x)(C[p := B] \Rightarrow D)}$$

and

$$\frac{D \Rightarrow (A \equiv B)}{(\forall x)(D \Rightarrow C[p := A]) \equiv (\forall x)(D \Rightarrow C[p := B])} \tag{1}$$

We prove the "weak" "full-capture" versions (2) and (3).

It is obvious that we cannot do any better: Full-capture "strong" versions will yield strong generalization!

$$\vdash A \equiv B \text{ implies } \vdash (\forall x)(C[p \setminus A] \Rightarrow D) \equiv (\forall x)(C[p \setminus B] \Rightarrow D) \tag{2}$$

and

$$\vdash A \equiv B \text{ implies } \vdash (\forall x)(D \Rightarrow C[p \setminus A]) \equiv (\forall x)(D \Rightarrow C[p \setminus B]) \tag{3}$$

Now, implication (2) is an instance of WLUS, where, without loss of generality, p occurs only in C. So it holds.

(3) has an identical proof. But what happened to the $D \Rightarrow$-part on the premise side? It was dropped, because *the rule is invalid with it* (see also [To]).

Indeed, take $D = x \approx 0$, $C = (\forall x)p$, $A = x \approx 0$ and $B = \mathit{true}$. Then,

$$\models D \Rightarrow (A \equiv B)$$

that is

$$\models x \approx 0 \Rightarrow (x \approx 0 \equiv \mathit{true})$$

but

$$\not\models (\forall x)(D \Rightarrow C[p \setminus A]) \equiv (\forall x)(D \Rightarrow C[p \setminus B])$$

that is

$$\not\models (\forall x)(x \approx 0 \Rightarrow (\forall x)x \approx 0) \equiv (\forall x)(x \approx 0 \Rightarrow (\forall x)\mathit{true})$$

As an aside, (1) *is* valid if the premise has an absolute proof:

$$\vdash D \Rightarrow (A \equiv B)$$

To prove the conclusion of (1), establish instead

$$\vdash D \Rightarrow (C[p := A]) \equiv C[p := B]) \tag{4}$$

To this end, assume D. This yields $A \equiv B^{\dagger}$ (by modus ponens and the premise of (1)). By SLCS, $C[p := A] \equiv C[p := B]$ follows, and hence so does (4) (Deduction Theorem).

Now, the formula in (4) yields

$$\vdash (D \Rightarrow C[p := A]) \equiv (D \Rightarrow C[p := B]) \tag{5}$$

which (Tautology Theorem) yields

$$\vdash (D \Rightarrow C[p := A]) \Rightarrow (D \Rightarrow C[p := B]) \tag{6}$$

† *Not* as an absolute theorem!

and

$$\vdash (D \Rightarrow C[p := A]) \Leftarrow (D \Rightarrow C[p := B]) \tag{7}$$

Since both (6) and (7) are absolute theorems, MON—on the I-Form $(\forall x)*$—and the Tautology Theorem conclude the argument.

Note that we cannot do any better: If (1) is taken literally ("strongly"), then it yields the invalid in E-logic strong generalization $A \vdash (\forall x)A$ (take $D = B = true$, $C = p$ in (1)).

Acknowledgements. I wish to thank my colleague Jonathan Ostroff for the many illuminating discussions we have had about Equational Logic.

8. Bibliography

[Ba] Barwise, J. "An introduction to first-order logic", in *Handbook of Mathematical Logic* (J. Barwise, Ed.), 5–46, Amsterdam: North-Holland Publishing Company, 1978.

[Bou] Bourbaki, N. *Élémens de Mathématique; Théorie des Ensembles*, **Ch. 1**, Paris: Hermann, 1966.

[DSc] Dijkstra, E. W., and Scholten, C. S. B. *Predicate Calculus and Program Semantics*, New York: Springer-Verlag, 1990.

[En] Enderton, H. B. *A mathematical introduction to logic*, New York: Academic Press, 1972.

[Gr] Gries, D. *Foundations for Calculational Logic*, Mathematical methods in program development (Marktoberdorf, 1996), 83–126, NATO Adv. Sci. Inst. Ser. f Comput. Systems Sci., 158, Berlin: Springer-Verlag, 1997.

[GS1] Gries, D. and Schneider, F. B. *A Logical Approach to Discrete Math*, New York: Springer-Verlag, 1994.

[GS2] Gries, D. and Schneider, F. B. *Equational propositional logic*, Information Processing Letters, **53**, 145–152, 1995.

[GS3] Gries, D. and Schneider, F. B. *Formalizations of substitution of equals for equals*, Pre-print, Sept. 1998 (personal communication).

[Man] Manin, Yu. I. *A Course in Mathematical Logic*, New York: Springer-Verlag, 1977.

[Men] Mendelson, E. *Introduction to mathematical logic*, 3rd Edition, Monterey, Calif: Wadsworth & Brooks, 1987.

[Schü] Schütte, K. *Proof Theory*, Berlin, Heidelberg, New York: Springer-Verlag, 1977.

[Sh] Shoenfield, J. R. *Mathematical Logic*, Reading, Mass.: Addison-Wesley, 1967.

[To] Tourlakis, G. *On the soundness and completeness of Equational Predicate Logics*, Department of Computer Science TR CS-1998-08, York University, November 1998.