Explicit provability: the intended semantics for intuitionistic and modal logic *

Sergei N. Artemov †

September, 1998

Abstract

The intended meaning of intuitionistic logic is given by the Brouwer-Heyting-Kolmogorov (*BHK*) semantics which informally defines intuitionistic truth as provability and specifies the intuitionistic connectives via operations on proofs. The natural problem of formalizing the *BHK* semantics and establishing the completeness of propositional intuitionistic logic $\mathcal{I}nt$ with respect to this semantics remained open until recently. This question turned out to be a part of the more general problem of the intended semantics for Gödel's modal logic of provability $\mathcal{S}4$ with the atoms "F is provable" which was open since 1933. In this paper we present complete solutions to both of these problems.

We find the logic of explicit provability ($\mathcal{LP}$) with the atoms "t is a proof of F" and establish that every theorem of $\mathcal{S}4$ admits a reading in $\mathcal{LP}$ as the statement about explicit provability. This provides the adequate provability semantics for $\mathcal{S}4$ along the lines of a suggestion made by Gödel in 1938. The explicit provability reading of Gödel's embedding of $\mathcal{I}nt$ into $\mathcal{S}4$ gives the desired formalization of the *BHK* semantics: $\mathcal{I}nt$ is shown to be complete with respect to this semantics. In addition, $\mathcal{LP}$ has revealed the relationship between proofs and types, and subsumes the λ-calculus, modal λ-calculus and combinatory logic.

1 Intended provability semantics for intuitionistic logic

According to Brouwer, intuitionistic truth means provability: "a statement is true if we have a proof of it, and false if we can show that the assumption that there is a proof for the statement leads to a contradiction" ([72], p.4). This semantics is implicit in some of Brouwer's papers, e.g. [16]. In 1930 A. Heyting suggested the axiom system $\mathcal{I}nt$ for intuitionistic logic ([28])[1].

*Technical Report CFIS 98-10, Cornell University

†Department of Mathematics, Cornell University, `email:artemov@math.cornell.edu` and Moscow University, Russia.

[1] The name $\mathcal{I}nt$ will signify propositional intuitionistic logic.

In 1931-34 Heyting and Kolmogorov made Brouwer's definition of intuitionistic truth explicit, though informal, by introducing what is now known as *Brouwer-Heyting-Kolmogorov (BHK) semantics*. *BHK* semantics is widely recognized as the intended semantics for intuitionistic logic ([18],[19],[20],[24],[37],[47],[50],[72],[73],[74],[75],[76]). *BHK* semantics gives an informal explanation of the truth of intuitionistic connectives. A statement is true if it has a proof, and a proof of a logically compound statement is given in terms of the proofs of its components. The description uses the unexplained primitive notions of *construction* and *proof*.

- A proof of a proposition $A \wedge B$ consists of a proof of A and a proof of B,
- a proof of $A \vee B$ is given by presenting either a proof of A or a proof of B,
- a proof of $A \rightarrow B$ is a construction which, given a proof of A returns a proof of B,
- absurdity $\perp$ is a proposition which has no proof and a proof of $\neg A$ is a construction which, given a proof of A, would return a proof of $\perp$.

This semantics was partially introduced by Heyting [29] (clauses for conjunction and disjunction), and by Kolmogorov [34] (clauses for implication and negation). The above formulation of *BHK* semantics appeared in [30]. For further comments one may consult [18],[20],[24], [69],[72],[73],[74].

The natural problem of formalizing *BHK* semantics and establishing the completeness of $\mathcal{I}nt$ with respect to this semantics remained open until recently despite a long history of studies in this area (see section 3 of this paper).

> To be sure, there are many models of different natures known for $\mathcal{I}nt$. A semantics for $\mathcal{I}nt$ is *adequate* if $\mathcal{I}nt$ is (sound and) complete with respect to this semantics. A number of adequate semantics for intuitionistic logic have been found: algebraic (Birkhof, [11]), topological (McKinsey-Tarski, [48]), Kripke semantics ([41]), and some others. Algebraic models for $\mathcal{I}nt$ are given by pseudo-boolean algebras, which generalizes the boolean algebra semantics of classical logic. Topological semantics for $\mathcal{I}nt$ is similar to set-theoretical semantics for classical logic. In a given topological space propositional variables are evaluated by arbitrary subsets, conjunction and disjunction operate in the usual set-theoretical manner, while intuitionistic implication and negation operate as classical implication and negation followed by the *interior* operation. Kripke model for $\mathcal{I}nt$ is a collection of the usual $0-1$ evaluations of atomic propositions (*possible worlds*) connected by a reflexive and transitive binary *accessibility relation* and satisfying *knowledge preservation principle*: if a statement holds in some world, then it also holds in all the worlds accessible from the given one. Again, in every world the truth of conjunction or disjunction is determined according to the usual classical truth tables. Implication

or negation is true in a world iff it is true classically in every world accessible from the given one. Comprehensive surveys of these and other semantics for intuitionistic logic can be found in [18],[61],[72].

BHK semantics gave rise to intensive studies of constructive semantics for intuitionistic theories, first of all realizability. The basic notions of realizability were defined along the lines of *BHK* clauses with different constructive objects instead of proofs: computable functions and their codes (e.g. in [32],[33]), computable operations of higher types (e.g. in [38]), partial recursive operations (e.g. in [21],[22]), etc. For the references one may consult recent surveys on realizability and constructive semantics [8],[71].

Note that the standard realizability semantics for $\mathcal{I}nt$ is not adequate. First of all, following Kleene ([32]) one should distinguish between intuitionistic and classical understanding of realizability semantics for intuitionistic theories. Intuitionistic realizability enjoys some nice completeness properties but does not provide an independent semantics for $\mathcal{I}nt$. For example, as follows from [58], a formula F is provable in intuitionistic predicate logic iff all arithmetical instances of F are provably realizable in a certain extension $\mathbf{HA}^+$ of intuitionistic arithmetic. Such a result relates $\mathcal{I}nt$ with a formal theory based on the same $\mathcal{I}nt$ and thus is not intended to give an independent semantics for the latter. On the other hand, classical realizabilities (Kleene realizability [32], function realizability [33], modified realizability [38], Medvedev's calculus of finite problems [50] and its variants), give conditions necessary but not sufficient for $\mathcal{I}nt$(cf.[18],[71],[74],[75]).

It turned out that the natural deduction proofs for $\mathcal{I}nt$ can be transliterated by the Curry-Howard isomorphism into the language of typed λ-terms (see, for example, [24],[20],[72]). The inductive definition of the Curry-Howard isomorphism goes along the lines of *BHK* clauses, where λ-terms play the role of *BHK* proofs. Though very important for establishing connections between derivations/formulas of $\mathcal{I}nt$ and terms/types in λ-calculus, a Curry-Howard presentation does not give an independent semantical characterization for $\mathcal{I}nt$. Indeed, under this presentation the realization of a sentence is modulo to isomorphism a derivation of this sentence in the same $\mathcal{I}nt$. Loosely speaking, from the *BHK* semantics perspective, the Curry-Howard isomorphism provides a trivial solution: a formula F is true, by definition, if F is derivable in $\mathcal{I}nt$.

2 Classical vs. intuitionistic *BHK* semantics

Despite strong similarities between Heyting's and Kolmogorov's descriptions of the provability semantics for $\mathcal{I}nt$, their approaches had fundamentally different objectives.

Heyting explained propositional intuitionistic logic $\mathcal{I}nt$ in terms of the intuitionistic understanding of constructions and proofs. His semantics gives a partial analysis of the intuitionistic meaning of a statement and does not intend to provide a foundation for $\mathcal{I}nt$ independent of the intuitionistic assumptions.

Kolmogorov in [34] intended to interpret $\mathcal{I}nt$ on the basis of the usual mathematical notion of problem solution (e.g., proof), and thus to provide a *definition* of intuitionistic logic within classical mathematics. Kolmogorov suggested reading $\mathcal{I}nt$ as the calculus of solvable schemes of problems. The basic notions of Kolmogorov's interpretation are problems and problem solutions. Each proposition denotes a problem. Solutions of the compound problems are described in terms of the solutions of their components along the lines of the *BHK* clauses above (reading "proof" as "solution"). A problem scheme $A(\vec{p})$ is *solved*, if there exists a general method of solving the problem A for any particular choice of the problems $\vec{p}$ and their solutions. Kolmogorov noticed that all axioms of the Heyting calculus for propositional intuitionistic logic $\mathcal{I}nt$ stood for the solved problem schemes, the rules preserved the property of a scheme being solved, and thus all schemes derived in $\mathcal{I}nt$ were solved. Kolmogorov also assumed implicitly that all such schemes could be derived from the Heyting axioms for $\mathcal{I}nt$ and therefore $\mathcal{I}nt$ was the calculus of the solved problem schemes. In his comments [35] of 1985 Kolmogorov elaborates:

> "The paper [34] was written in a hope that the logic of solutions of problems would eventually become a permanent part of a logic course. It was supposed to create a unified logical technique dealing with two types of objects: statements and problems."[2]

This difference between the Heyting and Kolmogorov semantics for $\mathcal{I}nt$ was acknowledged by Heyting himself in [30]. A. Troelstra in [70] characterized Kolmogorov's interpretation of $\mathcal{I}nt$ as "meaningful independently of intuitionistic bias."

Since the authors of the name "*BHK semantics*" were apparently aware of the differences between the Heyting and Kolmogorov approaches, we do not suggest changing this well established name. However, for the purposes of formalization of *BHK* semantics it is important to distinguish between classical and intuitionistic interpretations of *BHK* clauses. We suggest the name *classical BHK semantics* for the former and *intuitionistic BHK semantics* for the latter. Thus, Kolmogorov's reading of $\mathcal{I}nt$ as the logic of problem solutions may be considered classical *BHK* semantics.

A mathematical explication of intuitionistic *BHK* semantics would depend on a choice of intuitionistic theory to take *BHK* proofs from. Eventually, it would lead to an interpretation of $\mathcal{I}nt$ in a system based on $\mathcal{I}nt$ and presumably more complicated than $\mathcal{I}nt$. Such a semantics

[2] Translated from Russian by SA.

could not provide an independent foundation for intuitionistic logic. We will not address the issue of intuitionistic *BHK* semantics in this paper.

We demonstrate that classical *BHK* semantics, in turn, admits an exact mathematical formalization, which indeed provides an adequate semantics for $\mathcal{I}nt$ on the basis of the usual classical notion of proof.

3 Semantics of $\mathcal{I}nt$ via modal provability logic

Probably the first paper on formal provability semantics for intuitionistic logic was written in 1928 by Orlov ([57]). He introduced a unary logical connective (we call this connective $\Box$, for the sake of notational uniformity) with the informal reading of $\Box F$ as "F is provable". Orlov suggested prefixing all subformulas of a given propositional intuitionistic formula by $\Box$, and understanding the logical connectives in the usual classical way. Orlov's modal axioms for provability coincide with the ones for the modal logic $\mathcal{S}4$, which was later recognized as the modal logic for provability ([25]). Orlov used a certain proper fragment of classical logic in the background, thus making his system weaker than $\mathcal{S}4$. Nevertheless, he succeeded in deducing a number of properties of the provability operator and reproducing some basic laws of intuitionistic logic, e.g. $\neg\neg\neg a \leftrightarrow \neg a$.

Apparently independent of [57], Gödel in 1933 introduced the modal logic of provability and explicitly defined $\mathcal{I}nt$ in this logic. Gödel's provability logic has the same modal axioms and rules as the one from [57], i.e.

- $\Box F \rightarrow F$,
- $\Box(F \rightarrow G) \rightarrow (\Box F \rightarrow \Box G)$,
- $\Box F \rightarrow \Box\Box F$,
- $F \vdash \Box F$ (*necessitation* rule),

admits all axioms and rules of classical logic, and therefore coincides with the classical modal logic $\mathcal{S}4$. Gödel considered the translation $t(F)$ of an intuitionistic formula F into the classical modal language similar to the one from [57]: "box each subformula of F". Gödel established that

$$\mathcal{I}nt \vdash F \qquad \Rightarrow \qquad \mathcal{S}4 \vdash t(F),$$

thus providing an exact reading of the $\mathcal{I}nt$ formulas as statements about provability in classical mathematics. He conjectured that the inverse $\Leftarrow$ also holds. This conjecture was eventually established in [49].

However, the ultimate goal of defining $\mathcal{I}nt$ via the notion of a proof in classical mathematics had not been achieved because $\mathcal{S}4$ was left without an exact intended semantics of the provability operator $\Box$. Gödel himself was the first who addressed the issue of provability semantics for $\mathcal{S}4$ ([25], cf.[70]). He pointed out that the straightforward reading of $\Box F$ as "F is provable in a certain formal system" contradicted his incompleteness theorem.

> Let us consider first order arithmetic $\mathcal{PA}$. Let $\perp$ be the boolean constant *false*; then the $\mathcal{S}4$-axiom $\Box\perp \rightarrow \perp$ corresponds to the statement *Consis* $\mathcal{PA}$, expressing consistency of $\mathcal{PA}$. By necessitation, $\mathcal{S}4$ derives $\Box(\Box\perp \rightarrow \perp)$. The latter formula expresses the assertion that *Consis* $\mathcal{PA}$ is provable in $\mathcal{PA}$, which is false according to the second Gödel incompleteness theorem.

In [26] (cf.[59]) Gödel again acknowledged the problem of the provability semantics for $\mathcal{S}4$. This issue was also addressed by Lemmon [44], Myhill [55],[56], Kripke [40], Montague [54], Mints [52], Kuznetsov & Muravitsky [43], Goldblatt [27], Boolos [12],[14] Shapiro [62],[63], Buss [17], Artemov [1], and many others. However, the problem of finding an adequate provability semantics for $\mathcal{S}4$ has remained open.

A principal difficulty here is caused by the existential quantifier over proofs in the provability formula $\mathit{Provable}(y)$, which is $\exists x \mathit{Proof}(x,y)$, where $\mathit{Proof}(x,y)$ is the standard arithmetical formula saying "x is the code of a proof of a formula with the code y". The formula $\mathit{Provable}(y)$ may be characterized as the *implicit provability operator*, since in a model of arithmetic $\mathit{Provable}(F)$ does not always guarantee the existence of a proof of F. Indeed, in a given model of $\mathcal{PA}$ an element that instantiates the existential quantifier in $\exists x \mathit{Proof}(x,F)$ may be nonstandard. In this case $\exists x \mathit{Proof}(x,F)$ (i.e. $\mathit{Provable}(F)$) is true in the model, but there is no "real" $\mathcal{PA}$-derivation behind such an x. This explains why the reflection principle $\mathit{Provable}(F)\rightarrow F$ is not derivable in $\mathcal{PA}$: the formula $\mathit{Provable}(F)$ does not necessarily deliver a "real" proof of F.

This consideration suggests the idea of introducing a kind of explicit provability logic by switching from the formulas $\exists x \mathit{Proof}(x,F)$ to the formulas *Proof(t,F)* and replacing the existential quantifier on proofs in the former by Skolem style operations on proofs in the latter. The usual Skolem technique, however, does not work here, since there are no uniform commutation laws for the quantifiers and the provability operator.

> Some of these operations appeared in the proof of Gödel's second incompleteness theorem. Within that proof (cf.[12],[14]),[51],[65]) in order to establish what are now known as Hilbert-Bernays-Löb derivability conditions one constructs computable functions $m(x,y)$ and $c(x)$ such that
>
> $$\mathcal{PA} \vdash \mathit{Proof}(s, F\rightarrow G) \wedge \mathit{Proof}(t,F) \;\rightarrow\; \mathit{Proof}(m(s,t),G),$$
>
> $$\mathcal{PA} \vdash \mathit{Proof}(t,F) \rightarrow \mathit{Proof}(c(t), \mathit{Proof}(t,F)).$$

Then those facts are relaxed to their simplified versions

$$\mathcal{PA} \vdash Provable(F \to G) \wedge Provable(F) \ \to \ Provable(G),$$

$$\mathcal{PA} \vdash Provable(F) \to Provable(Provable(F)),$$

sufficient to establish the incompleteness theorem.

In one of his lectures [26] in 1938 (first published in 1995, see also [59]) Gödel sketched an explicit version of $\mathcal{S}4$ [3] with the basic proposition "t is a proof of F" and operations similar to $m(x,y)$ and $c(x)$. Although this sketch does not contain exact definitions, it shows the way to explain the reflexivity principle for provability logic, which was the major difficulty in $\mathcal{S}4$.

Gödel's proposal generalized the problem of formalization of classical *BHK* semantics for $\mathcal{I}nt$ to the problem of building an explicit provability logic: presumably, the former was derivable from the latter. The questions about an appropriate language and a complete set of axioms for explicit provability logic, as well as the question about its ability to realize $\mathcal{I}nt$ and $\mathcal{S}4$ had remained open.

Kreisel in [37],[39] (apparently without knowledge of [26]) developed a formal theory of constructions with a basic predicate like Gödel's "t is a proof of F", but with only partial success (cf.[59],[72],[76]).

In this paper we present a recent solution of the following problems, discussed above.

1. *To give the intended semantics and to find a complete axiom system for the explicit provability logic sketched by Gödel in 1938 ([26]).*

We consider the logical language in Gödel's format "t is a proof of F" and give its exact provability semantics. We demonstrate that one more operation should be added to Gödel's sketch of the explicit provability logic in order to enable it to emulate the entirety of $\mathcal{S}4$. We call the resulting system the *Logic of Proofs* ($\mathcal{LP}$)[4]. Here we establish the soundness and completeness of $\mathcal{LP}$ with respect to the intended provability semantics (Theorem 7.1).

2. *To find an adequate provability semantics for the Gödel provability logic $\mathcal{S}4$ ([25]).*

We establish that $\mathcal{LP}$ realizes all of $\mathcal{S}4$ by assigning proof terms to the modalities in every $\mathcal{S}4$-derivation (Theorem 8.2). This gives an adequate provability model for $\mathcal{S}4$ along the lines of Gödel's suggestion in [26].

[3]Gödel's sketch was rather clear about the propositional principles of explicit provability logic. It also mentioned possible principles involving the first order quantifiers, but was not specific on this matter. We consider the propositional part of Gödel's sketch only.

[4]$\mathcal{LP}$ was found by the author independently of Gödel's paper [26]. The first presentations of $\mathcal{LP}$ took place at the author's talks at the conferences in Münster and Amsterdam in 1994. Preliminary versions of $\mathcal{LP}$ along with the completeness theorem and realization of $\mathcal{S}4$ in $\mathcal{LP}$ appeared in Technical Reports [4], [6], cf. also a survey [31]. Note that despite its title the paper [3] does not introduce $\mathcal{LP}$.

3. *To formalize the classical BHK semantics for $\mathcal{I}nt$ and to establish the completeness of intuitionistic logic with respect to this semantics.*

We consider two realizations of $\mathcal{I}nt$ in $\mathcal{LP}$. The first one is defined by Gödel's translation of intuitionistic formulas into modal language "box all subformulas", with the subsequent realization in $\mathcal{LP}$. The second one is the McKinsey-Tarski translation ("box all atoms and implications") followed by the realization in $\mathcal{LP}$. Each of those two semantics is established to be adequate for intuitionistic propositional logic. This confirms Kolmogorov's assumption of 1932 that intuitionistic logic $\mathcal{I}nt$ coincides with the calculus of solutions to problems in classical mathematics. $\mathcal{LP}$ may be considered as the "unified logical technique dealing with two types of objects: statements and problems" meant by Kolmogorov in 1932 ([34],[35]). This also achieves the original objective of Gödel (1933) to define $\mathcal{I}nt$ via the classical notion of proof.

$\mathcal{LP}$ provides a provability semantics for certain areas of logic and applications where main objects have had informal provability interpretations. For example, $\mathcal{LP}$ may be considered as a generalization of combinatory logic in that it is able to iterate the type assignment ':'. In particular, $\mathcal{LP}$ can express the propositions of the form $t:(s:F)$, which are outside the scope of the usual combinatory logic. $\mathcal{LP}$ naturally contains the defined abstraction operator λ^*x which is an extension of the defined λ-abstraction operator λ^*x in combinatory logic (cf.[73]). This generalizes the Curry-Howard presentation of intuitionistic proofs as typed λ-terms. Moreover, through realizations in $\mathcal{LP}$ both modality and λ-terms receive a uniform provability semantics and thus may be treated as the objects of the same nature, namely proof terms.

4 Logic of Proofs

4.1 Definition. The language of Logic of Proofs ($\mathcal{LP}$) contains

the usual language of classical propositional logic
proof variables $x_0, \ldots, x_n, \ldots$, proof constants $a_0, \ldots, a_n, \ldots$
function symbols: monadic !, binary $\cdot$ and $+$
operator symbol of the type "*term : formula*".

We will use $a, b, c, \ldots$ for proof constants, $u, v, w, x, y, z, \ldots$ for proof variables, i, j, k, l, m, n for natural numbers. Terms are defined by the grammar

$$p ::= x_i \mid a_i \mid !p \mid p_1 \cdot p_2 \mid p_1 + p_2$$

We call these terms *proof polynomials* and denote them by $p,r,s,t\ldots$. By analogy we refer to constants as coefficients. Constants correspond to proofs of a finite fixed set of propositional schemas. We will also omit $\cdot$ whenever it is safe. We also assume that $(a \cdot b \cdot c)$, $(a \cdot b \cdot c \cdot d)$, *etc.* should be read as $((a \cdot b) \cdot c)$, $(((a \cdot b) \cdot c) \cdot d)$, *etc.*

Using t to stand for any term and S for any propositional letter, the formulas are defined by the grammar

$$\sigma ::= S \mid \sigma_1 \rightarrow \sigma_2 \mid \sigma_1 \wedge \sigma_2 \mid \sigma_1 \vee \sigma_2 \mid \neg\sigma \mid t{:}\sigma$$

We will use $A, B, C, F, G, H, X, Y, Z$ for the formulas in this language, and $\Gamma, \Delta, \dots$ for the finite sets (also finite multisets, or finite lists) of formulas unless otherwise explicitly stated. We will also use $\vec{x}, \vec{y}, \vec{z}, \dots$ and $\vec{p}, \vec{r}, \vec{s}, \dots$ for vectors of proof variables and proof polynomials respectively. If $\vec{s} = (s_1, \dots, s_n)$ and $\Gamma = (F_1, \dots, F_n)$, then $\vec{s}{:}\Gamma$ denotes $(s_1{:}F_1, \dots, s_n{:}F_n)$, $\bigvee\Gamma = F_1 \vee \dots \vee F_n$, $\bigwedge\Gamma = F_1 \wedge \dots \wedge F_n$. We assume the following precedences from highest to lowest: $!, \cdot, +, :, \neg, \wedge, \vee, \rightarrow$. We will use the symbol $=$ in different situations, both formal and informal. Symbol $\equiv$ denotes syntactical identity, $\ulcorner E \urcorner$ is the Gödel number of E.

The intended semantics for $p{:}F$ is "p is a proof of F", which will be formalized in the next section. Note that proof systems which provide a formal semantics for $p{:}F$ are *multi-conclusion* ones, i.e. p may be a proof of several different F's (see Comment 4.8).

4.2 Definition. The system $\mathcal{LP}_0$. Axioms:

A0. Finite set of axiom schemes of classical propositional logic in the language of $\mathcal{LP}$
A1. $t{:}F \rightarrow F$ *"verification"*
A2. $t{:}(F \rightarrow G) \rightarrow (s{:}F \rightarrow (t{\cdot}s){:}G)$ *"application"*
A3. $t{:}F \rightarrow\, !t{:}(t{:}F)$ *"proof checker"*
A4. $s{:}F \rightarrow (s{+}t){:}F, \quad t{:}F \rightarrow (s{+}t){:}F$ *"choice"*

Rule of inference:

R1. $$\frac{F \rightarrow G \qquad F}{G}$$ *"modus ponens".*

The system $\mathcal{LP}$ is $\mathcal{LP}_0$ plus the rule

R2. $$\frac{}{c{:}\mathbf{A}}\,,$$
if $\mathbf{A}$ *is an axiom A0 – A4, and c a proof constant* *"axiom necessitation".*

A *Constant Specification (CS)* is a finite set of formulas $c_1 : A_1, \dots, c_n : A_n$ such that c_i is a constant, and A_i an axiom *A0 – A4*. Each derivation in $\mathcal{LP}$ naturally generates the $\mathcal{CS}$ consisting of all formulas introduced in this derivation by the *axiom necessitation* rule.

4.3 Comment. Proof constants in $\mathcal{LP}$ stand for proofs of "simple facts", namely propositional axioms and axioms *A1 – A4*. In a way the proof constants resemble atomic constant terms (*combinators*) of typed combinatory logic (cf.[73]). A constant c_1 specified as

$c_1 : (A \to (B \to A))$ can be identified with the combinator $\mathbf{k}^{A,B}$ of the type $A \to (B \to A)$. A constant c_2 such that $c_2 : [(A \to (B \to C)) \to ((A \to B) \to (A \to C))]$ corresponds to the combinator $\mathbf{s}^{A,B,C}$ of the type $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$. The proof variables may be regarded as term variables of combinatory logic, the operation "$\cdot$" as the application of terms. In general an $\mathcal{LP}$-formula $t : F$ can be read as a combinatory term t of the type F. Typed combinatory logic $\mathbf{CL}_{\to}$ thus corresponds to a fragment of $\mathcal{LP}$ consisting only of formulas of the sort $t : F$ where t contains no operations other than "$\cdot$" and F is a formula built from the propositional letters by "$\to$" only.

There is no restriction on the choice of a constant c in *R2* within a given derivation. In particular, *R2* allows to introduce a formula $c : A(c)$, or to specify a constant several times as a proof of different axioms from *A0* – *A4*. One might restrict $\mathcal{LP}$ to injective constant specifications, i.e. only allowing each constant to serve as a proof of a single axiom $\mathbf{A}$ within a given derivation (although allowing constructions $c : \mathbf{A}(c)$, as before). Such a restriction would not change the ability of $\mathcal{LP}$ to emulate classical modal logic, or the functional and arithmetical completeness theorems for $\mathcal{LP}$ (below), though it will provoke an excessive renaming of the constants.

Both $\mathcal{LP}_0$ and $\mathcal{LP}$ enjoy the deduction theorem

$$\Gamma, A \vdash B \quad \Rightarrow \quad \Gamma \vdash A \to B,$$

and the substitution lemma: *If* $\Gamma(x, P) \vdash B(x, P)$ *for a propositional variable* P *and a proof variable* x*, then for any proof polynomial* t *and any formula* F

$$\Gamma(x/t, P/F) \vdash B(x/t, P/F).$$

For a given constant specification $\mathcal{CS}$ under $\mathcal{LP}(\mathcal{CS})$ we mean $\mathcal{LP}_0$ plus $\mathcal{CS}$. Obviously,

F *is derivable in* $\mathcal{LP}$ *with a constant specification* $\mathcal{CS}$ $\Leftrightarrow$ $\mathcal{LP}(\mathcal{CS}) \vdash F$ $\Leftrightarrow$ $\mathcal{LP}_0 \vdash \bigwedge \mathcal{CS} \to F$.

4.4 Proposition. (Lifting lemma) *Given a derivation* $\mathcal{D}$ *of the type*

$$\vec{s}{:}\Gamma, \Delta \vdash_{\mathcal{LP}} F,$$

one can construct a proof polynomial $t(\vec{x}, \vec{y})$ *such that*

$$\vec{s}{:}\Gamma, \vec{y}{:}\Delta \vdash_{\mathcal{LP}} t(\vec{s}, \vec{y}) : F.$$

Proof. By induction on the derivation $\vec{s}{:}\Gamma, \Delta \vdash F$. If $F = s_i : G_i \in \vec{s}{:}\Gamma$, then put $t := !s_i$ and use *A3*. If $F = D_j \in \Delta$, then put $t := y_j$. If F is an axiom *A0* – *A4*, then pick a fresh

proof constant c and put $t := c$; by *R2*, $\vdash c{:}F$. Let F be introduced by *modus ponens* from $G \to F$ and G. Then, by the induction hypothesis, there are proof polynomials $u(\vec{s},\vec{y})$ and $v(\vec{s},\vec{y})$ such that $u:(G \to F)$ and $v:G$ are both derivable in $\mathcal{LP}$ from $\vec{s}:\Gamma,\vec{y}:\Delta$. By *A2*, $\vec{s}{:}\Gamma,\vec{y}{:}\Delta \vdash (uv){:}F$, and we put $t := uv$. If F is introduced by *R2*, then $F = c{:}A$ for some axiom A. Use the same *R2* followed by *A3*: $c{:}A \to !c{:}c{:}A$, to get $\vec{s}{:}\Gamma,\vec{y}{:}\Delta \vdash !c{:}F$, and put $t := !c$.
◀

Note that if $\Delta \vdash_{\mathcal{LP}_0} F$, then one can construct $t(\vec{y})$ which is a product of proof constants and variables from $\vec{y}$ such that $\vec{y}:\Delta \vdash_{\mathcal{LP}_0} t(\vec{y}):F$. It is easy to see from the proof that the lifting polynomial $t(\vec{x},\vec{y})$ is nothing but a blueprint of $\mathcal{D}$. Thus $\mathcal{LP}$ accommodates its own proofs as terms.

4.5 Corollary. (Necessitation rule)

$$\vdash F \quad \Rightarrow \ \vdash p{:}F \ \ \textit{for some proof polynomial } p$$

This is a special case of lifting. It follows from the proof of lifting Lemma 4.4 that p here is a blueprint of a derivation of F in $\mathcal{LP}$ that is implicitly present in the assertion "$\vdash F$". Note, that p is a ground proof polynomial (i.e. p has no proof variables), which does not contain '+'.

As we can see in section 8 $\mathcal{LP}$ suffices to emulate all $\mathcal{S}4$-derivations.

4.6 Example. $\mathcal{S}4 \vdash (\Box A \wedge \Box B) \to \Box(A \wedge B)$

In $\mathcal{LP}$ the corresponding derivation is

1. $A \to (B \to A \wedge B)$, by *A0*,
2. $c{:}(A \to (B \to A \wedge B))$, from 1, by *R2*,
3. $x{:}A \to (c \cdot x){:}(B \to A \wedge B)$, from 2, by *A2*,
4. $x{:}A \to (y{:}B \to (c \cdot x \cdot y){:}(A \wedge B))$, from 3, by *A2* and propositional logic,
5. $x{:}A \wedge y{:}B \to (c \cdot x \cdot y){:}(A \wedge B))$, from 4, by propositional logic.

4.7 Example. $\mathcal{S}4 \vdash (\Box A \vee \Box B) \to \Box(A \vee B)$.

In $\mathcal{LP}$ the corresponding derivation is

1. $A \to A \vee B, \quad B \to A \vee B$, by *A0*,
2. $a{:}(A \to A \vee B), \quad b{:}(B \to A \vee B)$, by *R2*,
3. $x{:}A \to (a{\cdot}x){:}(A \vee B), \quad y{:}B \to (b{\cdot}y){:}(A \vee B)$, from 2, by *A2*,
4. $(a{\cdot}x){:}(A \vee B) \to (a{\cdot}x + b{\cdot}y){:}(A \vee B), \quad (b{\cdot}y){:}(A \vee B) \to (a{\cdot}x + b{\cdot}y){:}(A \vee B)$, by *A4*,
5. $(x{:}A \vee y{:}B) \to (a{\cdot}x + b{\cdot}y){:}(A \vee B)$, from 4, by propositional logic.

4.8 Comment. The operations "$\cdot$" and "!" are present for single-conclusion as well as on multi-conclusion proof systems. On the other hand, "+" is an operation for multi-conclusion proof systems only. Indeed, by *A4* we have $s\!:\!F \wedge t\!:\!G \rightarrow\; (s{+}t)\!:\!F \wedge (s{+}t)\!:\!G$, thus $s + t$ proves different formulas. The differences between single-conclusion and multi-conclusion proof systems are mostly cosmetic. Usual proof systems (Hilbert or Gentzen style) may be considered as single-conclusion if one assumes that a proof derives only the end formula (sequent) of a proof tree. On the other hand, the same systems may be regarded as multi-conclusion by assuming that a proof derives all formulas assigned to the nodes of the proof tree. The logic of strictly single-conclusion proof systems was studied in [2], [3] and in [42], where it meets a complete axiomatization (system $\mathcal{FLP}$). $\mathcal{FLP}$ is not compatible with any modal logic (cf. Comment 8.5) and thus is not directly relevant to the problem of finding an intended semantics for the modal logic of provability. Therefore, provability as a modal operator corresponds to multi-conclusion proof systems.

No single operator "$t\!:$" in $\mathcal{LP}$ is a normal modality since none of them satisfies the property $t\!:\!(P \rightarrow Q) \rightarrow (t\!:\!P \rightarrow t\!:\!Q)$. This makes $\mathcal{LP}$ essentially different from numerous polymodal logics, e.g. the dynamic logic of programs ([36]), where the modality is upgraded by some additional features. In turn, in the Logic of Proofs the modality is decomposed into a family of proof polynomials (see section 8).

5 Standard provability interpretation of $\mathcal{LP}$

The Logic of Proofs is meant to play for the notion of proof a role similar to that played by the boolean propositional logic for the notion of statement. It is shown in sections 5 and 7 of this paper that $\mathcal{LP}$ enjoys the soundness/completeness property:

$$\mathcal{LP} \vdash F \qquad \Leftrightarrow \qquad F \textit{ is true under any interpretation }.$$

Any system of proofs with a proof checker operation capable of internalizing its own proofs as terms (cf.[66]) may be within the scope of $\mathcal{LP}$. In particular, any proof system for first order Peano Arithmetic $\mathcal{PA}$ (cf.[12], [14], [51], [68]) provides a model for $\mathcal{LP}$ with Gödel numbers of proofs being an instrument for internalizing proofs as terms. The soundness ($\Rightarrow$) does not necessarily refer to arithmetical models. However, $\mathcal{PA}$ is convenient for establishing the completeness ($\Leftarrow$) of $\mathcal{LP}$: given $\mathcal{LP} \not\vdash F$ one can always find a proof system for $\mathcal{PA}$ along with an evaluation of variables in F which makes F false (Theorem 7.1).

In sections 5 and 7 of this paper by Δ_1 and Σ_1 we mean the corresponding classes of arithmetical predicates. We will use φ, ψ to denote arithmetical formulas, f, g, h to denote arithmetical terms, and i, j, k, l, n to denote natural numbers unless stated otherwise. We will use the letters u, v, w, x, y, z to denote individual variables in arithmetic and hope that a reader is able to distinguish them from the proof variables. If n is a natural number, then

$\overline{n}$ will denote a numeral corresponding to n, i.e. a standard arithmetical term $0'''^{\cdots}$ where $'$ is a successor functional symbol and the number of $'$'s equals n. We will use the simplified notation n for a numeral $\overline{n}$ when it is safe.

5.1 Definition. We assume that $\mathcal{PA}$ contains terms for all primitive recursive functions (cf. [68]), called *primitive recursive terms.* Formulas of the form $f(\vec{x}) = 0$ where $f(\vec{x})$ is a primitive recursive term are *standard primitive recursive formulas. A standard* Σ_1 *formula* is a formula $\exists x\varphi(x, \vec{y})$ where $\varphi(x, \vec{y})$ is a standard primitive recursive formula. An arithmetical formula φ is *provably* Σ_1 if it is provably equivalent in $\mathcal{PA}$ to a standard Σ_1 formula; φ is *provably* Δ_1 iff both φ and $\neg\varphi$ are provably Σ_1.

5.2 Definition. A *proof predicate* is a provably Δ_1-formula $Prf(x, y)$ such that for every arithmetical sentence φ

$$\mathcal{PA} \vdash \varphi \quad \Leftrightarrow \quad \text{for some } n \in \omega \quad Prf(n, \ulcorner\varphi\urcorner) \text{ holds}^5.$$

A proof predicate *Prf(x,y)* is *normal* if the following conditions are fulfilled:

1) (*finiteness of proofs*) For every proof k the set $T(k) = \{l \mid Prf(k, l)\}$ is finite. The function from k to the canonical number of $T(k)$ is computable.

2) (*conjoinability of proofs*) For any natural numbers k and l there is a natural number n such that

$$T(k) \cup T(l) \subseteq T(n).$$

The conjoinability indicates that normal proof predicates are multi-conclusion ones.

5.3 Comment. Every normal proof predicate can be transformed into a single-conclusion one by changing from

$$\text{“}p \text{ proves } F_1, \ldots, F_n\text{”} \qquad \text{to} \qquad \text{“}(p, i) \text{ proves } F_i,\ i = 1, \ldots, n\text{”}.$$

In turn, every single-conclusion proof predicate may be regarded as normal multi-conclusion by reading

$$\text{“}p \text{ proves } F_1 \wedge \ldots \wedge F_n\text{”} \qquad \text{as} \qquad \text{“}p \text{ proves each of } F_i,\ i = 1, \ldots, n\text{”}.$$

5.4 Proposition. *For every normal proof predicate Prf there are computable functions $m(x, y)$, $a(x, y)$, $c(x)$ such that for all arithmetical formulas φ, ψ and all natural numbers k, n the following formulas are valid:*

[5] We have omitted bars over numerals for natural numbers $n, \ulcorner\varphi\urcorner$ in the formula *Prf* and will do it consistently throughout this paper.

$$Prf(k,\ulcorner\varphi\to\psi\urcorner)\wedge Prf(n,\ulcorner\varphi\urcorner)\to Prf(m(k,n),\ulcorner\psi\urcorner)$$
$$Prf(k,\ulcorner\varphi\urcorner)\to Prf(a(k,n),\ulcorner\varphi\urcorner),\qquad Prf(n,\ulcorner\varphi\urcorner)\to Prf(a(k,n),\ulcorner\varphi\urcorner)$$
$$Prf(k,\ulcorner\varphi\urcorner)\to Prf(c(k),\ulcorner Prf(k,\ulcorner\varphi\urcorner)\urcorner).$$

Proof. The following function can be taken as m:

> *Given k,n set $m(k,n)=\mu z.$"$Prf(z,\ulcorner\psi\urcorner)$ for all ψ such that there are $\ulcorner\varphi\to\psi\urcorner\in T(k)$ and $\ulcorner\varphi\urcorner\in T(n)$" .*

Likewise, for a one could take

> *Given k,n set $a(k,n)=\mu z.$ "$T(k)\cup T(n)\subseteq T(z)$".*

Finally, c may be given by

> *Given k set $c(k)=\mu z.$"$Prf(z,\ulcorner Prf(k,\ulcorner\varphi\urcorner)\urcorner)$ for all $\ulcorner\varphi\urcorner\in T(k)$". Such a z always exists. Indeed, $Prf(k,\ulcorner\varphi\urcorner)$ is a true Δ_1 sentence for every $\ulcorner\varphi\urcorner\in T(k)$, therefore they all are provable in $\mathcal{PA}$. Use conjoinability to find a uniform proof of all of them.*

◀

Note that the natural arithmetical proof predicate *PROOF(x,y)*

"*x is the code of a derivation containing a formula with the code y*".

is an example of a normal proof predicate.

5.5 Definition. An arithmetical *interpretation* $*$ of the $\mathcal{LP}$-language has the following parameters:

- a normal proof predicate *Prf* with the functions $m(x,y)$, $a(x,y)$, $c(x)$ as in Proposition 5.4,
- an evaluation of propositional letters by sentences of arithmetic, and
- an evaluation of proof variables and proof constants by natural numbers.

Let $*$ commute with boolean connectives,

$$(t\cdot s)^*=m(t^*,s^*),\quad (t+s)^*=a(t^*,s^*),\quad (!t)^*=c(t^*),$$
$$(t\!:\!F)^*=Prf(\overline{t^*},\overline{\ulcorner F^*\urcorner}).$$

Under an interpretation $*$ a proof polynomial t becomes the natural number t^*, an $\mathcal{LP}$-formula F becomes the arithmetical sentence F^*. A formula $(t{:}F)^*$ is always provably Δ_1. Note that $\mathcal{PA}$ (as well as any theory containing a certain finite set of arithmetical axioms, e.g. Robinson's arithmetic) is able to derive any true Δ_1 sentence, and thus to derive a negation of any false Δ_1 sentence (cf.[51]). For a set X of $\mathcal{LP}$-formulas under X^* we mean the set of all F^*'s such that $F \in X$. Given a constant specification $\mathcal{CS}$, an arithmetical interpretation $*$ is a *$\mathcal{CS}$-interpretation* if all formulas from $\mathcal{CS}^*$ are true (equivalently, are provable in $\mathcal{PA}$). An $\mathcal{LP}$-formula F is *valid* (with respect to the arithmetical semantics) if the arithmetical formula F^* is true under all interpretations $*$. F is *$\mathcal{CS}$-valid* if F^* is true under all $\mathcal{CS}$-interpretations $*$.

5.6 Proposition. (Arithmetical soundness of $\mathcal{LP}_0$)

1. *If $\mathcal{LP}_0 \vdash F$ then F is valid.*
2. *If $\mathcal{LP}_0 \vdash F$ then $\mathcal{PA} \vdash F^*$ for any interpretation $*$.*

Proof. A straightforward induction on the derivation in $\mathcal{LP}_0$. Let us check 2. for the axiom $t : F \to F$. Under an interpretation $*$ $(t{:}F \to F)^* \equiv \mathit{Prf}(t^*, \ulcorner F^* \urcorner) \to F^*$. Consider two possibilities. Either $\mathit{Prf}(t^*, \ulcorner F^* \urcorner)$ is true, in which case t^* is indeed a proof of F^*, thus $\mathcal{PA} \vdash F^*$ and $\mathcal{PA} \vdash (t{:}F \to F)^*$. Otherwise $\mathit{Prf}(t^*, \ulcorner F^* \urcorner)$ is false, in which case being a false Δ_1 formula it is refutable in $\mathcal{PA}$, i.e. $\mathcal{PA} \vdash \neg \mathit{Prf}(t^*, \ulcorner F^* \urcorner)$ and again $\mathcal{PA} \vdash (t{:}F \to F)^*$.
◀

5.7 Corollary. (Arithmetical soundness of $\mathcal{LP}$)

1. *If $\mathcal{LP}(\mathcal{CS}) \vdash F$ then F is $\mathcal{CS}$-valid.*
2. *If $\mathcal{LP}(\mathcal{CS}) \vdash F$ then $\mathcal{PA} \vdash F^*$ for any $\mathcal{CS}$-interpretation $*$.*

5.8 Comment. The standard provability semantics for $\mathcal{LP}$ above may be characterized as a *call-by-value* semantics, since the evaluation F^* of a given $\mathcal{LP}$-formula F depends upon the value of participating functions. A *call-by-name* provability semantics for $\mathcal{LP}$ was introduced in [4] and then used in [42], [64]. In the latter semantics F^* depends upon the particular programs for the functions participating in $*$.

In order to define the *call-by-name* provability semantics for $\mathcal{LP}$ we assume that $\mathcal{PA}$ has the standard set of tools to introduce ι-terms. We use a new functional symbol $\iota z.\varphi(z)$ for each arithmetical formula $\varphi(z)$ and assume that ι-terms could be eliminated by using the small scope convention (cf.[20]). The term $\iota z.\varphi(z)$ is called *computable* if $\varphi(z)$ is provably Σ_1. A computable term represents some computable function, every computable function is represented by a computable term (cf.[51]).

The term $\iota z.\varphi(z)$ is *provably total* if $\mathcal{PA} \vdash \exists_1 z\varphi(z)$, i.e. $\mathcal{PA}$ proves that there exists a unique z such that $\varphi(z)$. In particular, every arithmetical term in a narrow sense, i.e. a term built from 0 by $', +, \times$ may be regarded as a provably total computable term. *A closed computable term* is a computable provably total term $\iota z.\varphi(z)$ such that $\varphi(z)$ contains no free variables other than z.

The set of computable terms is closed under substitution. The result of substituting a closed computable term into a Δ_1 formula is again a Δ_1 formula. Closed computable terms stand for all computable "names" for natural numbers. There is an algorithm which for any closed computable term f calculates its *value*, i.e. the numeral $\overline{n}$ such that $\mathcal{PA} \vdash f = \overline{n}$.

An analog of Proposition 5.4 can be established that for every normal proof predicate Prf there are computable terms $m(x,y)$, $a(x,y)$, $c(x)$ such that if f, g are closed computable terms, then $m(f,g)$, $a(f,g)$, $c(\ulcorner f \urcorner)$ are again closed computable terms and for all arithmetical formulas φ, ψ the following formulas are valid:

$$\mathit{Prf}(f, \ulcorner \varphi \to \psi \urcorner) \wedge \mathit{Prf}(g, \ulcorner \varphi \urcorner) \to \mathit{Prf}(m(f,g), \ulcorner \psi \urcorner)$$

$$\mathit{Prf}(f, \ulcorner \varphi \urcorner) \to \mathit{Prf}(a(f,g), \ulcorner \varphi \urcorner), \qquad \mathit{Prf}(g, \ulcorner \varphi \urcorner) \to \mathit{Prf}(a(f,g), \ulcorner \varphi \urcorner)$$

$$\mathit{Prf}(f, \ulcorner \varphi \urcorner) \to \mathit{Prf}(c(\ulcorner f \urcorner), \ulcorner \mathit{Prf}(f, \ulcorner \varphi \urcorner) \urcorner).$$

Note that $c(\ulcorner f \urcorner)$ depends on the code of f rather than on the value of f. In particular, it may be the case that the values of f and g are equal, but $c(\ulcorner f \urcorner) \neq c(\ulcorner g \urcorner)$.

An interpretation $*$ is defined by the choice of a normal proof predicate Prf with the terms $m(x,y)$, $a(x,y)$, $c(x)$, an evaluation of propositional letters by sentences of arithmetic, and an evaluation of proof variables and proof constants by closed computable terms. As before $*$ commutes with boolean connectives, $(t \cdot s)^* = m(t^*, s^*)$, $(t+s)^* = a(t^*, s^*)$, $(!t)^* = c(\ulcorner t^* \urcorner)$, $(t\!:\!F)^* = \mathit{Prf}(t^*, \ulcorner F^* \urcorner)$. Note that unlike the standard call-by-value interpretation above in this case we substitute not the numeral of the value of f for the variable x in $\mathit{Prf}(x,y)$ but a term f itself. Under any interpretation $*$ a proof polynomial t becomes a closed computable term t^*, an $\mathcal{LP}$-formula F becomes an arithmetical sentence F^*. A formula $(t\!:\!F)^*$ is always provably Δ_1.

As it was established in [4] $\mathcal{LP}$ is sound and complete with respect to this call-by-name provability interpretation. In fact the soundness in this case can be shown by an easy modification of the soundness proof for the standard call-by-name interpretation above. In Comment 7.15 we will discuss how to establish the completeness of $\mathcal{LP}$ in the call-by-name case.

6 A sequent formulation of Logic of Proofs

By *sequent* we mean a pair $\Gamma \Rightarrow \Delta$, where Γ and Δ are finite multisets of $\mathcal{LP}$-formulas. For Γ, F we understand $\Gamma \cup \{F\}$.

Axioms of $\mathcal{LPG}_0$ are sequents of the form $\Gamma, F \Rightarrow F, \Delta$ and $\Gamma, \bot \Rightarrow \Delta$. Along with the usual Gentzen sequent rules of classical propositional logic, including the cut and construction rules (e.g. like **G2c** from [73]), the system $\mathcal{LPG}_0$ contains the rules

$$\frac{A, \Gamma \Rightarrow \Delta}{t{:}A, \Gamma \Rightarrow \Delta}\ (:\ \Rightarrow) \qquad\qquad \frac{\Gamma \Rightarrow \Delta, t{:}A}{\Gamma \Rightarrow \Delta, !t{:}t{:}A}\ (\Rightarrow !)$$

$$\frac{\Gamma \Rightarrow \Delta, t{:}A}{\Gamma \Rightarrow \Delta, (t+s){:}A}\ (\Rightarrow +) \qquad\qquad \frac{\Gamma \Rightarrow \Delta, t{:}A}{\Gamma \Rightarrow \Delta, (s+t){:}A}\ (\Rightarrow +)$$

$$\frac{\Gamma \Rightarrow \Delta, s{:}(A \to B) \qquad \Gamma \Rightarrow \Delta, t{:}A}{\Gamma \Rightarrow \Delta, (s \cdot t){:}B}\ (\Rightarrow \cdot)$$

As will follow from the proof of 7.1 the rule $(\Rightarrow \cdot)$ for $\mathcal{LPG}_0$ (but not for $\mathcal{LPG}$) can in fact be limited by the condition that $A \to B$ must occur in Γ, Δ, without losing any provable sequents.

The system $\mathcal{LPG}$ is $\mathcal{LPG}_0$ plus the rule

$$\frac{}{\Gamma \Rightarrow c{:}\mathbf{A}, \Delta}\ (\Rightarrow c),$$

where $\mathbf{A}$ is an axiom *A0 – A4* from section 4, and c is a proof constant.

$\mathcal{LPG}^-$ and $\mathcal{LPG}_0^-$ are the corresponding systems without the rule Cut.

6.1 Proposition. $\mathcal{LPG}_0 \vdash \Gamma \Rightarrow \Delta$ *iff* $\mathcal{LP}_0 \vdash \bigwedge \Gamma \to \bigvee \Delta$, $\mathcal{LPG} \vdash \Gamma \Rightarrow \Delta$ *iff* $\mathcal{LP} \vdash \bigwedge \Gamma \to \bigvee \Delta$.

The proof proceeds by a straightforward induction both ways.

6.2 Corollary. $\mathcal{LP}(\mathcal{CS}) \vdash F$ *iff* $\mathcal{LPG}_0 \vdash \mathcal{CS} \Rightarrow F$.

6.3 Definition. The sequent $\Gamma \Rightarrow \Delta$ is *saturated* if

1. $A \to B \in \Gamma$ implies $B \in \Gamma$ or $A \in \Delta$,
2. $A \to B \in \Delta$ implies $A \in \Gamma$ and $B \in \Delta$[6],

[6]The clauses concerning other boolean connectives are optional.

3. $t{:}A \in \Gamma$ implies $A \in \Gamma$,
4. $!t{:}t{:}A \in \Delta$ implies $t{:}A \in \Delta$,
5. $(s+t){:}A \in \Delta$ implies $s{:}A \in \Delta$ and $t{:}A \in \Delta$
6. $(s \cdot t) : B \in \Delta$ implies *for each* $X \to B$ *occurring as a subformula in* Γ, Δ *either* $s{:}(X \to B) \in \Delta$ *or* $t{:}X \in \Delta$.

6.4 Lemma. (Saturation lemma) *Suppose* $\mathcal{LPG}_0^- \not\vdash \Gamma \Rightarrow \Delta$. *Then there exists a saturated sequent* $\Gamma' \Rightarrow \Delta'$ *such that*

1. $\Gamma \subseteq \Gamma'$, $\Delta \subseteq \Delta'$,
2. $\Gamma' \Rightarrow \Delta'$ *is not derivable in* $\mathcal{LPG}_0^-$.

Proof. A saturated sequent is obtained by the following *Saturation Algorithm* $\mathcal{SA}$. Given $\Gamma \Rightarrow \Delta$, for each undischarged formula S from $\Gamma \cup \Delta$ non-deterministically try to perform one of the following steps. At the moment 0 all formulas from $\Gamma \cup \Delta$ are available After a step is performed discharge S (make it unavailable). If none of the clauses 1 - 7 is applicable terminate with success.

1. if $S = (A \to B) \in \Gamma$, then put A into Δ or B into Γ,
2. if $S = (A \to B) \in \Delta$, then put A into Γ and B into Δ,
3. if $S = t{:}A \in \Gamma$, then put A into Γ,
4. if $S =!t{:}t{:}A \in \Delta$, then put $t{:}A$ into Δ,
5. if $S = (s+t){:}A \in \Delta$, then put both $s{:}A$ and $t{:}A$ into Δ,
6. if $S = (s \cdot t){:}B \in \Delta$, then for each $X_1, \ldots, X_n$ such that $X_i \to B$ is a subformula in Γ, Δ put either $s{:}(X_i \to B)$ or $t{:}X_i$ into Δ,
7. if $\Gamma \cap \Delta \neq \emptyset$ or $\bot \in \Gamma$, then backtrack. If backtracked to the root node terminate with failure. When backtracking to a given node make available again all the formulas discharged after leaving this node the previous time.

The Saturation Algorithm $\mathcal{SA}$ terminates. Indeed, $\mathcal{SA}$ is finitely branching and each non-backtracking step breaks either a subformula of $\Gamma \Rightarrow \Delta$ or a formula of the type $t{:}F$, where both t and F occur in $\Gamma \Rightarrow \Delta$. There are only finitely many of those formulas, which guarantees termination. Moreover, $\mathcal{SA}$ terminates with success. Indeed, otherwise $\mathcal{SA}$ terminates at the root node $\Gamma \Rightarrow \Delta$ of the computation tree with all the possibilities exhausted and no way to backtrack. Then the computation tree $\mathcal{T}$ of $\mathcal{SA}$ contains the sequent $\Gamma \Rightarrow \Delta$ at the root, and $\mathcal{LPG}_0$ axioms at the leaf nodes. By a standard induction on the depth of a node in $\mathcal{T}$ one can prove that every sequent in $\mathcal{T}$ is derivable in $\mathcal{LPG}_0^-$, which contradicts the assumption that $\mathcal{LPG}_0^- \not\vdash \Gamma \Rightarrow \Delta$. The nodes corresponding to the steps 1 – 5 and 7 are trivial. Let us consider a node which corresponds to 6. Such a node is labelled by a sequent $\Pi \Rightarrow \Theta, st{:}B$, and its children are 2^n sequents of the form $\Pi \Rightarrow \Theta, st{:}B, Y_1^\sigma, \ldots, Y_n^\sigma$, where $\sigma = (\sigma_1 \ldots, \sigma_n)$ is an n-tuple of 0's and 1's, and

$$Y_i^\sigma = \begin{cases} s\!:\!(X_i \to B), & \text{if } \sigma_i = 0 \\ t\!:\!X_i, & \text{if } \sigma_i = 1. \end{cases}$$

Here $X_1, \ldots, X_n$ is the list of all formulas such that $X_i \to B$ is a subformula of $\Gamma \Rightarrow \Delta$. By the induction hypothesis all the child sequents are derivable in $\mathcal{LPG}_0^-$. In particular, among them there are 2^{n-1} pairs of sequents of the form $\Pi \Rightarrow \Theta', s\!:\!(X_1 \to B)$ and $\Pi \Rightarrow \Theta', t\!:\!X_1$. To every such pair apply the rule $(\Rightarrow \cdot)$ to obtain $\Pi \Rightarrow \Theta'$ (we assume that $st : B \in \Theta'$). The resulting 2^{n-1} sequents are of the form $\Pi \Rightarrow \Theta, st\!:\!B, Y_2^\sigma, \ldots, Y_n^\sigma$. After we repeat this procedure $n-1$ more times we end up with the sequent $\Pi \Rightarrow \Theta, st\!:\!B$, which is thus derivable in $\mathcal{LPG}_0^-$.
◀

Note that in a saturated sequent $\Gamma \Rightarrow \Delta$ which is not $\mathcal{LPG}_0^-$-derivable the set Γ is closed under the rules $t\!:\!X/X$ and $X \to Y, X/Y$.

6.5 Lemma. *For each saturated sequent $\Gamma \Rightarrow \Delta$ not derivable in $\mathcal{LPG}_0^-$ there is a set of $\mathcal{LP}$-formulas $\widetilde{\Gamma}$ (a completion of $\Gamma \Rightarrow \Delta$) such that*

1. *$\widetilde{\Gamma}$ is a provably decidable set, for each term t the set $I(t) = \{X \mid t\!:\!X \in \widetilde{\Gamma}\}$ is finite and a function from a code[7] of t to a code[8] of $I(t)$ is provably computable,*
2. *$F \in \Gamma$ implies $F \in \widetilde{\Gamma}$, $\Delta \cap \widetilde{\Gamma} = \emptyset$,*
3. *if $t\!:\!X \in \widetilde{\Gamma}$, then $X \in \widetilde{\Gamma}$,*
4. *if $s\!:\!(X \to Y) \in \widetilde{\Gamma}$ and $t\!:\!X \in \widetilde{\Gamma}$, then $(s \cdot t)\!:\!Y \in \widetilde{\Gamma}$,*
5. *if $t\!:\!X \in \widetilde{\Gamma}$, then $!t\!:\!t\!:\!X \in \widetilde{\Gamma}$,*
6. *if $t\!:\!X \in \widetilde{\Gamma}$ and s is a proof polynomial, then $(t+s)\!:\!X \in \widetilde{\Gamma}$ and $(s+t)\!:\!X \in \widetilde{\Gamma}$.*

Proof. We describe a *completion algorithm* $\mathcal{COM}$ that produces a series of finite sets of $\mathcal{LP}$-formulas $\Gamma_0, \Gamma_1, \Gamma_2, \ldots$. Let $\Gamma_0 = \{F \mid F \in \Gamma\}$.

For each natural number $i > 1$ let $\mathcal{COM}$ do the following:

if $i = 3k$, then $\mathcal{COM}$ sets

$$\Gamma_{i+1} = \Gamma_i \bigcup_{s,t} \{(s \cdot t)\!:\!Y \mid s\!:\!(X \to Y), t\!:\!X \in \Gamma_i\},$$

if $i = 3k+1$, then $\mathcal{COM}$ sets

$$\Gamma_{i+1} = \Gamma_i \bigcup_{t} \{!t\!:\!t\!:\!X \mid t\!:\!X \in \Gamma_i\},$$

[7] For example, the Gödel number of t.

[8] For example, the canonical number of the finite set of Gödel numbers of formulas from $I(t)$.

if $i = 3k+2$, then $\mathcal{COM}$ sets

$$\Gamma_{i+1} = \Gamma_i \bigcup_{s,t} \{(s+t){:}X, (t+s){:}X \mid t{:}X \in \Gamma_i, |s| < i.\}^9$$

Let

$$\widetilde{\Gamma} = \bigcup_i \Gamma_i.$$

By definition, $\Gamma_i \subseteq \Gamma_{i+1}$.

It is easy to see that at step $i > 0$ $\mathcal{COM}$ produces either a formula from Γ or formulas of the form $t{:}X$ with the length of t greater than $i/3$. This observation secures the decidability of $\widetilde{\Gamma}$. Indeed, given a formula F of length n wait until step $i = 3n$ of $\mathcal{COM}$; $F \in \Gamma_n$ iff $F \in \widetilde{\Gamma}$. Similar argument establishes the decidability of $I(t)$ from which one can construct the desired provable computable arithmetical term for $I(t)$.

In order to establish 2 and 3 we prove by induction on i that for all $i = 0, 1, 2, \ldots$

A. $\Gamma_i \cap \Delta = \emptyset$,
B. $t{:}X \in \Gamma_i \quad \Rightarrow \quad X \in \Gamma_i$,
C. $X \to Y, X \in \Gamma_i \quad \Rightarrow \quad Y \in \Gamma_i$.

The base case $i = 0$ holds because of the saturation properties of $\Gamma_0 = \Gamma$.

For the induction step assume the induction hypothesis that the properties A,B, and C hold for i and consider Γ_{i+1}.

A. Suppose there is $F \in \Gamma_{i+1} \cap \Delta$ but $F \notin \Gamma_i$. There are three possibilities. If $i - 1 = 3k$ then F is $(s \cdot t){:}Y$ such that $s{:}(X \to Y), t{:}X \in \Gamma_i$ for some X. From the description of $\mathcal{COM}$ it follows that $(X \to Y) \in \Gamma$. By the saturation properties of $\Gamma \Rightarrow \Delta$, since $(s \cdot t){:}Y \in \Delta$ and $X \to Y$ occurs in Γ either $s{:}(X \to Y) \in \Delta$ or $t{:}X \in \Delta$. In either case $\Gamma_i \cap \Delta \neq \emptyset$ which is impossible by the induction hypothesis.

If $i - 1 = 3k+1$ then F is $!t{:}t{:}X$ such that $t{:}X \in \Gamma_i$. By the saturation properties of Δ, $t{:}X \in \Delta$. Again $\Gamma_i \cap \Delta \neq \emptyset$ which is impossible by the induction hypothesis.

If $i - 1 = 3k+2$ then F is $(t+s){:}X$ such that either $t{:}X \in \Gamma_i$ or $s{:}X \in \Gamma_i$. By the saturation properties, from $(t+s){:}X \in \Delta$ conclude that both $t{:}X \in \Delta$ and $s{:}X \in \Delta$. Once again, $\Gamma_i \cap \Delta \neq \emptyset$ which is impossible by the induction hypothesis.

Thus $\Gamma_{i+1} \cap \Delta = \emptyset$.

B. Suppose $p{:}B \in \Gamma_{i+1}$ and $p{:}B \notin \Gamma_i$. We conclude that in this case $B \in \Gamma_{i+1}$. Indeed, again there are three possibilities.

If If $i - 1 = 3k$ then $p{:}B$ is $(s \cdot t){:}Y$ such that $s{:}(X \to Y), t{:}X \in \Gamma_i$ for some X. By the induction hypothesis for Γ_i, $(X \to Y), X \in \Gamma_i$ and thus $Y \in \Gamma_i$. By the inclusion $\Gamma_i \subseteq \Gamma_{i+1}$, $Y \in \Gamma_{i+1}$.

[9] $|s|$ is the length of s, i.e. the total number of variables, constants, and functional symbols in s.

If $i-1=3k+1$ then $p{:}B$ is $!t{:}t{:}X$ such that $t{:}X\in\Gamma_i$. Then $t{:}X\in\Gamma_{i+1}$.

If $i-1=3k+2$ then $p{:}B$ is $(t+s){:}B$ such that either $t{:}B\in\Gamma_i$ or $s{:}B\in\Gamma_i$. By the induction hypothesis, in either case $B\in\Gamma_i$, therefore $B\in\Gamma_{i+1}$.

C. Suppose $X\rightarrow Y, X\in\Gamma_{i+1}$. From the description of $\mathcal{COM}$ it follows that $(X\rightarrow Y)\in\Gamma$. By the saturation properties of $\Gamma\Rightarrow\Delta$, either $Y\in\Gamma$ or $X\in\Delta$. In the former case we are done. If $X\in\Delta$ then $\Gamma_{i+1}\cap\Delta\neq\emptyset$, which is impossible by item A of the induction step.

Items 4., 5., and 6. of Lemma 6.5 are guaranteed by the definition of $\mathcal{COM}$. Indeed, if some *if* condition is fulfilled, then it occurs at step i and $\mathcal{COM}$ necessarily puts the *then* formula into Γ_{i+3} at the latest.

◀

7 Consolidated completeness theorem

In this section we establish completeness and cut elimination theorems for the Logic of Proofs.

7.1 Theorem. *The following are equivalent*

1. $\mathcal{LPG}_0^-\vdash\Gamma\Rightarrow\Delta$,
2. $\mathcal{LPG}_0\vdash\Gamma\Rightarrow\Delta$,
3. $\mathcal{LP}_0\vdash\bigwedge\Gamma\rightarrow\bigvee\Delta$,
4. *for every interpretation* $*$ $\mathcal{PA}\vdash(\bigwedge\Gamma\rightarrow\bigvee\Delta)^*$,
5. *for every interpretation* $*$ *the formula* $(\bigwedge\Gamma\rightarrow\bigvee\Delta)^*$ *is true.*

Proof. The steps from 1 to 2 and from 4 to 5 are trivial. The step from 2 to 3 follows from 6.1, and the step from 3 to 4 follows from 5.6. The only remaining step is thus from 5 to 1. We assume "not 1" and establish "not 5". Suppose $\mathcal{LPG}_0^-\nvdash\Gamma\Rightarrow\Delta$. Our aim now will be to construct an interpretation $*$ such that $(\bigwedge\Gamma\rightarrow\bigvee\Delta)^*$ is false (in the standard model of arithmetic).

From the saturation procedure get a saturated sequent $\Gamma'\Rightarrow\Delta'$ (6.4), and then make a completion to get a set of formulas $\widetilde{\Gamma}'$ (6.5).

We define the desired interpretation $*$ on propositional letters S_i, proof variables x_j and proof constants a_j first. We assume that Gödel numbering of the joint language of $\mathcal{LP}$ and $\mathcal{PA}$ is injective, i.e.

$$\ulcorner E_1\urcorner=\ulcorner E_2\urcorner\quad\leftrightarrow\quad E_1\equiv E_2$$

for any expressions E_1, E_2, and that 0 is not a Gödel number of any expression. For a propositional letter S, proof variable x and proof constant a let

$$S^*=\begin{cases}\ulcorner S\urcorner=\ulcorner S\urcorner, & \text{if } S\in\widetilde{\Gamma}'\\ \ulcorner S\urcorner=0, & \text{if } S\notin\widetilde{\Gamma}',\end{cases}\qquad x^*=\ulcorner x\urcorner,\qquad a^*=\ulcorner a\urcorner.$$

The remaining parts of $*$ are constructed by an arithmetical fixed point equation below.

For any arithmetical formula $\mathit{Prf}(x,y)$ define an auxiliary translation † of $\mathcal{LP}$-terms to numerals and $\mathcal{LP}$-formulas to $\mathcal{PA}$-formulas such that $S^\dagger = S^*$ for any propositional letter S, $t^\dagger = \ulcorner t \urcorner$ for any $\mathcal{LP}$-term t, $(t{:}F)^\dagger = \mathit{Prf}(t^\dagger, \ulcorner F^\dagger \urcorner)$, and † commutes with the propositional connectives.

It is clear that if $\mathit{Prf}(x,y)$ contains quantifiers, then † is injective, i.e. $F^\dagger \equiv G^\dagger$ yields $F \equiv G$. Indeed, from $F^\dagger \equiv G^\dagger$ it follows that the principal connectives in F and G coincide. We consider one case: $(F_1 \to F_2)^\dagger \equiv (s{:}G)^\dagger$ is impossible. Since $(s{:}G)^\dagger \equiv \mathit{Prf}(k,n)$ for the corresponding numerals k and n, this formula contains quantifiers. Therefore the formula $(F_1 \to F_2)^\dagger \equiv {F_1}^\dagger \to {F_2}^\dagger$ also contains quantifiers and thus contains a subformula of the form $\mathit{Prf}(k_1,n_1)$. However, $(s{:}G)^\dagger \equiv {F_1}^\dagger \to {F_2}^\dagger$ is impossible since the numbers of logical connectives and quantifiers in both parts of $\equiv$ are different. Now the injectivity of † can be shown by an easy induction on the construction of an $\mathcal{LP}$-formula. Moreover, one can construct primitive recursive functions f and g such that

$$f(\ulcorner B \urcorner, \ulcorner \mathit{Prf} \urcorner) = \ulcorner B^\dagger \urcorner, \quad g(\ulcorner B^\dagger \urcorner, \ulcorner \mathit{Prf} \urcorner) = \ulcorner B \urcorner.$$

Let *(PROOF, $\otimes$, $\oplus$, $\Uparrow$)* be the standard multi-conclusion proof predicate from section 5, with $\otimes$ standing for application, $\oplus$ for choice and $\Uparrow$ for proof checker operations on proofs associated with *PROOF*. In particular, for any arithmetical formulas φ, ψ and any natural numbers k, n the following formulas are true:

$$PROOF(k, \ulcorner \varphi \to \psi \urcorner) \wedge PROOF(n, \ulcorner \varphi \urcorner) \to \mathit{Prf}(k \otimes n, \ulcorner \psi \urcorner)$$

$$PROOF(k, \ulcorner \varphi \urcorner) \to PROOF(k \oplus n, \ulcorner \varphi \urcorner), \quad PROOF(n, \ulcorner \varphi \urcorner) \to PROOF(k \oplus n, \ulcorner \varphi \urcorner)$$

$$PROOF(k, \ulcorner \varphi \urcorner) \to PROOF({\Uparrow}k, \ulcorner PROOF(k, \ulcorner \varphi \urcorner) \urcorner).$$

For technical convenience and without loss of generality we assume that $PROOF(\ulcorner t \urcorner, k)$ is false for any $\mathcal{LP}$-term t and any $k \in \omega$.

By $\mu x.\varphi(x, \vec{y})$ we mean a function that calculates x such that

$$\varphi(x, \vec{y}) \wedge \forall z < x \neg \varphi(z, \vec{y}).$$

It is clear that $\mu x.\varphi(x, \vec{y})$ is computable if $\varphi(x, \vec{y}) \wedge \forall z < x \neg \varphi(z, \vec{y})$ is provably Σ_1. There are two convenient sufficient conditions under each of which $\mu x.\varphi(x, \vec{y})$ is computable:

$\varphi(x, \vec{y})$ is provably Δ_1,

$\varphi(x, \vec{y})$ is provably Σ_1 and functional with respect to x, i.e. $\varphi(k_1, \vec{n}) \wedge \varphi(k_2, \vec{n}) \to k_1 = k_2$ is true for all $k_1, k_2, \vec{n}$.

By an arithmetical fixed point argument we construct a formula $\mathit{Prf}(x,y)$ such that $\mathcal{PA}$ proves the following *fixed point equation (FPE)*:

$$\begin{array}{lll} Prf(x,y) & \leftrightarrow & PROOF(x,y) \;\;\vee \\ & & (\text{“}x = \ulcorner t \urcorner \textit{ for some } \mathcal{LP}\textit{-term } t \textit{ and} \\ & & y = \ulcorner B^{\dagger} \urcorner \textit{ for some } \mathcal{LP}\textit{-formula } B \textit{ such that } B \in I(t)\text{”}) \end{array}$$

Here the arithmetical formula "…" describes a primitive recursive procedure: given x and y recover t and B such that $x = \ulcorner t \urcorner$ and $y = \ulcorner B^{\dagger} \urcorner$, then verify $B \in I(t)$. From *FPE* it is immediate that *Prf* is a provably Δ_1-formula, since $PROOF(x,y)$ is provably Δ_1. It also follows from *FPE* that $\mathcal{PA} \vdash \psi$ yields $Prf(k, \ulcorner \psi \urcorner)$ for some $k \in \omega$.

We define the arithmetical formulas $M(x,y,z)$, $A(x,y,z)$, $C(x,z)$ as follows

$$\begin{array}{lll} M(x,y,z) & \leftrightarrow & (\text{“}x = \ulcorner s \urcorner \textit{ and } y = \ulcorner t \urcorner \textit{ for some } \mathcal{LP}\textit{-terms } s \textit{ and } t\text{”} \wedge z = \ulcorner s \cdot t \urcorner) \vee \\ & & \\ & & (\text{“}x = \ulcorner s \urcorner \textit{ for some } \mathcal{LP}\textit{-term } s \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } \mathcal{LP}\textit{-term } t\text{”} \wedge \\ & & \exists v[\text{“}v = \mu w.(\bigwedge\{PROOF(w, \ulcorner B^{\dagger} \urcorner) \mid B \in I(s)\})\text{”} \wedge z = v \otimes y]) \vee \\ & & \\ & & (\text{“}x \neq \ulcorner s \urcorner \textit{ for any } \mathcal{LP}\textit{-term } s \textit{ and } y = \ulcorner t \urcorner \textit{ for some } \mathcal{LP}\textit{-term } t\text{”} \wedge \\ & & \exists v[\text{“}v = \mu w.(\bigwedge\{PROOF(w, \ulcorner B^{\dagger} \urcorner) \mid B \in I(t)\})\text{”} \wedge z = x \otimes v]) \vee \\ & & \\ & & (\text{“}x \neq \ulcorner s \urcorner \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } \mathcal{LP}\textit{-terms } s \textit{ and } t\text{”} \wedge z = x \otimes y) \end{array}$$

$$\begin{array}{lll} A(x,y,z) & \leftrightarrow & (\text{“}x = \ulcorner s \urcorner \textit{ and } y = \ulcorner t \urcorner \textit{ for some } \mathcal{LP}\textit{-terms } s \textit{ and } t\text{”} \wedge z = \ulcorner s + t \urcorner) \vee \\ & & \\ & & (\text{“}x = \ulcorner s \urcorner \textit{ for some } \mathcal{LP}\textit{-term } s \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } \mathcal{LP}\textit{-term } t\text{”} \wedge \\ & & \exists v[\text{“}v = \mu w.(\bigwedge\{PROOF(w, \ulcorner B^{\dagger} \urcorner) \mid B \in I(s)\})\text{”} \wedge z = v \oplus y]) \vee \\ & & \\ & & (\text{“}x \neq \ulcorner s \urcorner \textit{ for any } \mathcal{LP}\textit{-term } s \textit{ and } y = \ulcorner t \urcorner \textit{ for some } \mathcal{LP}\textit{-term } t\text{”} \wedge \\ & & \exists v[\text{“}v = \mu w.(\bigwedge\{PROOF(w, \ulcorner B^{\dagger} \urcorner) \mid B \in I(t)\})\text{”} \wedge z = x \oplus v]) \vee \\ & & \\ & & (\text{“}x \neq \ulcorner s \urcorner \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } \mathcal{LP}\textit{-terms } s \textit{ and } t\text{”} \wedge z = x \oplus y) \end{array}$$

$$\begin{array}{ll} C(x,z) \leftrightarrow & (\text{“}x = \ulcorner t \urcorner \textit{ for some } \mathcal{LP}\textit{-term } t\text{”} \wedge z = \ulcorner !t \urcorner) \vee \\ & (\text{“}x \neq \ulcorner t \urcorner \textit{ for any } \mathcal{LP}\textit{-term } t\text{”} \wedge \\ & \exists v[\text{“}v = \mu w.(\bigwedge\{PROOF(w, \ulcorner PROOF(t, \ulcorner \varphi \urcorner) \to Prf(t, \ulcorner \varphi \urcorner) \urcorner) \mid \varphi \in T(t)\})\text{”} \wedge \\ & z = v \otimes \Uparrow x]) \end{array}$$

Here "…" denotes a natural arithmetical formula representing in $\mathcal{PA}$ the condition '…', "$v = \mu w.\psi$" is a natural formula representing in $\mathcal{PA}$ the function $\mu w.\psi$. Note that in the definitions above all these functions are computable since all the corresponding ψ's are provably Δ_1. Therefore $M(x,y,z)$, $A(x,y,z)$ and $C(x,z)$ are provably Σ_1. Moreover, these formulas are functional with respect to z. By the necessary conditions above the functions $m(x,y)$, $a(x,y)$ and $c(x)$ are computable.

We continue defining the interpretation $*$. Let *Prf* for $*$ be the one from *FPE*,

$$m(x,y) := \mu z.M(x,y,z), \quad a(x,y) := \mu z.A(x,y,z), \quad c(x) := \mu z.C(x,z).$$

7.2 Lemma.
a) $t^* = t^\dagger$ for any $\mathcal{LP}$-term t,
b) $B^* \equiv B^\dagger$ for any $\mathcal{LP}$-formula B.

Proof. a) Induction on the construction of an $\mathcal{LP}$-term. Base cases are covered by the definition of the interpretation $*$. For the induction step note that according to the definitions, the following equalities are provable in $\mathcal{PA}$:

$$(s\cdot t)^* = m(s^*,t^*) = m(\ulcorner s\urcorner,\ulcorner t\urcorner) = \ulcorner s\cdot t\urcorner = (s\cdot t)^\dagger,$$

$$(s+t)^* = a(s^*,t^*) = a(\ulcorner s\urcorner,\ulcorner t\urcorner) = \ulcorner s+t\urcorner = (s+t)^\dagger,$$

$$(!t)^* = c(t^*) = c(\ulcorner t\urcorner) = \ulcorner !t\urcorner = (!t)^\dagger.$$

b) By an induction on B we prove that B^* and $B^\dagger$ coincide. The atomic case when B is a propositional letter holds by the definitions. If B is $t:F$, then $(t:F)^* = Prf(t^*,\ulcorner F^*\urcorner)$. By a), $t^* = t^\dagger$. By the induction hypothesis, $F^* \equiv F^\dagger$ which yields $\ulcorner F^*\urcorner = \ulcorner F^\dagger\urcorner$. Therefore $Prf(t^*,\ulcorner F^*\urcorner) = Prf(t^\dagger,\ulcorner F^\dagger\urcorner) = (t:F)^\dagger$. The inductive steps are trivial.
◀

7.3 Corollary. *The mapping $*$ is injective on terms and formulas of $\mathcal{LP}$. In particular, for all expressions E_1 and E_2*

$${E_1}^* = {E_2}^* \quad\Rightarrow\quad E_1 \equiv E_2.$$

7.4 Corollary. *X^* is provably Δ_1 for any $\mathcal{LP}$-formula X .*

Indeed, if X is atomic, then X is provably Δ_1 by the definition of $*$. If X is $t:Y$, then $(t:Y)^*$ is $Prf(t^*,\ulcorner Y^*\urcorner)$. By Lemma 7.2,

$$\mathcal{PA} \vdash Prf(t^*,\ulcorner Y^*\urcorner) \leftrightarrow Prf(\ulcorner t\urcorner,\ulcorner Y^*\urcorner).$$

The latter formula is provably Δ_1, therefore $(t:Y)^*$ is provably Δ_1. Since the class of provably Δ_1 formulas is closed under boolean connectives X^* is provably Δ_1 for each X.

7.5 Lemma. *If $X \in \widetilde{\Gamma'}$, then $\mathcal{PA} \vdash X^*$, if $X \in \Delta'$, then $\mathcal{PA} \vdash \neg X^*$.*

Proof. By induction on the length of X. Base case, i.e. X is atomic or $X = t:Y$. Let X be atomic. By the definition of $*$, X^* is true iff $X \in \widetilde{\Gamma'}$. Let $X = t:Y$ and $t:Y \in \widetilde{\Gamma'}$. Then $\mathcal{PA} \vdash$ "$Y \in I(t)$". By *FPE*, $\mathcal{PA} \vdash \mathit{Prf}(\ulcorner t \urcorner, \ulcorner Y^\dagger \urcorner)$. By Lemma 7.2, $\mathcal{PA} \vdash \mathit{Prf}(t^*, \ulcorner Y^* \urcorner)$. Therefore $\mathcal{PA} \vdash (t:Y)^*$.

If $t:Y \in \Delta'$, then $t:Y \notin \widetilde{\Gamma'}$ and "$Y \in I(t)$" is false. The formula $PROOF(t^*, \ulcorner Y^* \urcorner)$ is also false since t^* is $\ulcorner t \urcorner$ (by Lemma 7.2) and $PROOF(\ulcorner t \urcorner, k)$ is false for any k by assumption. By *FPE*, $(t:Y)^*$ is false. Since $(t:Y)^*$ is provably Δ_1 (Lemma 7.4) $\mathcal{PA} \vdash \neg(t:Y)^*$.

The induction steps corresponding to boolean connectives are standard and based on the saturation properties of $\Gamma' \Rightarrow \Delta'$. For example, let $X = Y \to Z \in \widetilde{\Gamma'}$. Then $Y \to Z \in \Gamma'$, and by Definition 6.3, $Y \in \Gamma'$ or $Z \in \Delta'$. By the induction hypothesis, Y^* is true or Z^* is false, and thus $(Y \to Z)^*$ is true, etc.
◀

7.6 Lemma. $\mathcal{PA} \vdash \varphi \quad \Leftrightarrow \quad \mathit{Prf}(n, \ulcorner \varphi \urcorner)$ *for some* $n \in \omega$.

Proof. It remains to establish ($\Leftarrow$). Let $\mathit{Prf}(n, \ulcorner \varphi \urcorner)$ for some $n \in \omega$. By *FPE* ,

$$\mathit{Prf}(n, \ulcorner \varphi \urcorner) \;\Rightarrow\; PROOF(n, \ulcorner \varphi \urcorner) \;\text{ or }\; \ulcorner \varphi \urcorner = \ulcorner B^\dagger \urcorner \;\text{ for some } B \text{ such that } t:B \in \widetilde{\Gamma'}.$$

In the latter case by the saturation property of $\widetilde{\Gamma'}$, $B \in \widetilde{\Gamma'}$. By Lemma 7.5, $\mathcal{PA} \vdash B^*$. By the injectivity of the Gödel numbering, $\varphi \equiv B^\dagger$. By Lemma 7.2, $\varphi \equiv B^*$. Therefore $\mathcal{PA} \vdash \varphi$.
◀

7.7 Lemma. *For all arithmetical formulas φ, ψ and natural numbers k, n the following is true*

a) $\mathit{Prf}(k, \ulcorner \varphi \to \psi \urcorner) \wedge \mathit{Prf}(n, \ulcorner \varphi \urcorner) \to \mathit{Prf}(m(k,n), \ulcorner \psi \urcorner)$

b) $\mathit{Prf}(k, \ulcorner \varphi \urcorner) \to \mathit{Prf}(a(k,n), \ulcorner \varphi \urcorner), \qquad \mathit{Prf}(n, \ulcorner \varphi \urcorner) \to \mathit{Prf}(a(k,n), \ulcorner \varphi \urcorner)$

c) $\mathit{Prf}(k, \ulcorner \varphi \urcorner) \to \mathit{Prf}(c(k), \ulcorner \mathit{Prf}(k, \ulcorner \varphi \urcorner) \urcorner)$.

Proof. *a)* Assume $\mathit{Prf}(k, \ulcorner \varphi \to \psi \urcorner)$ and $\mathit{Prf}(n, \ulcorner \varphi \urcorner)$. There are four possibilities.

i) Neither of k, n is equal to a Gödel number of an $\mathcal{LP}$-term. By *FPE*, both $PROOF(n, \ulcorner \varphi \urcorner)$ and $PROOF(k, \ulcorner \varphi \to \psi \urcorner)$ hold, so $PROOF(k \otimes n, \ulcorner \psi \urcorner)$ does also.

ii) Both k and n are equal to Gödel numbers of some $\mathcal{LP}$-terms, say $k = \ulcorner s \urcorner$ and $n = \ulcorner t \urcorner$. By *FPE*, φ is F^* and ψ is G^* for some $\mathcal{LP}$-formulas F, G such that $F \to G \in I(s)$ and $F \in I(t)$.

By the closure property of $\widetilde{\Gamma'}$ (Lemma 6.5(4)), $G \in I(s \cdot t)$. By *FPE*, $Prf(\ulcorner s \cdot t \urcorner, \ulcorner G^* \urcorner)$. By Lemma 7.2 and by definitions, $\mathcal{PA}$ proves that

$$\ulcorner s \cdot t \urcorner = (s \cdot t)^* = m(s^*, t^*) = m(\ulcorner s \urcorner, \ulcorner t \urcorner) = m(k, n).$$

Thus $m(k,n) = \ulcorner s \cdot t \urcorner$ and $Prf(m(k,n), \ulcorner \psi \urcorner)$ is true.

iii) k is not equal to the Gödel number of an $\mathcal{LP}$-term, $n = \ulcorner t \urcorner$ for some $\mathcal{LP}$-term t. By *FPE*, $PROOF(k, \ulcorner \varphi \to \psi \urcorner)$ and $\varphi \equiv F^\dagger$ for some $\mathcal{LP}$-formula F such that $F \in I(t)$. Compute the number

$$l = \mu w.(\bigwedge\{PROOF(w, \ulcorner B^\dagger \urcorner) \mid B \in I(t)\})$$

by the following method. Take $I(t) = \{B_1, \ldots, B_l\}$. By definition, $B_i \in \widetilde{\Gamma'}$, $i = 1, \ldots, l$. By Lemma 7.5, $\mathcal{PA} \vdash {B_i}^*$ for all $i = 1, \ldots, l$. By Lemma 7.2, $\mathcal{PA} \vdash {B_i}^\dagger$ for all $i = 1, \ldots, l$. By the conjoinability property of *PROOF* there exists w such that $PROOF(w, \ulcorner {B_i}^\dagger \urcorner)$ for all $i = 1, \ldots, l$. Let j be the least such w. In particular, $PROOF(j, \ulcorner F^\dagger \urcorner)$. By the definition of $\otimes$, $PROOF(k \otimes j, \ulcorner \psi \urcorner)$. By the definition of M, $\mathcal{PA} \vdash m(k,n) = k \otimes j$, therefore $PROOF(m(k,n), \ulcorner \psi \urcorner)$ holds.

Case iv): "s is a Gödel number of an $\mathcal{LP}$-term but t is not a Gödel number of any $\mathcal{LP}$-term" is similar to (iii).

Case (b) can be checked in the same way as (a).

c) Given $Prf(k, \ulcorner \varphi \urcorner)$ there are two possibilities.

i) $k = \ulcorner t \urcorner$ for some $\mathcal{LP}$-term t. By *FPE*, $\varphi \equiv F^\dagger$ for some F such that $F \in I(t)$. By the closure property 6.5(5) of $\widetilde{\Gamma'}$, $!t{:}t{:}F \in \widetilde{\Gamma'}$. By Lemma 7.5, $(!t{:}t{:}F)^*$ holds. By definitions,

$$(!t{:}t{:}F)^* \equiv Prf(c(t^*), \ulcorner Prf(t^*, \ulcorner F^* \urcorner) \urcorner).$$

By Lemma 7.2, $t^* = \ulcorner t \urcorner$ and $F^* \equiv F^\dagger$. Therefore $t^* = k$, $F^* \equiv \varphi$ and

$$Prf(c(k), \ulcorner Prf(k, \ulcorner \varphi \urcorner) \urcorner).$$

ii) $k \neq \ulcorner t \urcorner$ for any $\mathcal{LP}$-term t. By *FPE*, $PROOF(k, \ulcorner \varphi \urcorner)$ holds. By definition of the proof checking operation $\Uparrow$ for *PROOF*,

$$PROOF(\Uparrow k, \ulcorner PROOF(k, \ulcorner \varphi \urcorner) \urcorner).$$

By the definition of C, in this case $\mathcal{PA} \vdash c(k) = l \otimes \Uparrow k$ where

$$l = \mu w. \bigwedge\{PROOF(w, \ulcorner PROOF(k, \ulcorner \psi \urcorner) \to Prf(k, \ulcorner \psi \urcorner) \urcorner) \mid PROOF(k, \ulcorner \psi \urcorner)\}.$$

By the definition of l,

$$PROOF(l, \ulcorner PROOF(k, \ulcorner \varphi \urcorner) \to Prf(k, \ulcorner \varphi \urcorner) \urcorner).$$

Therefore

$$PROOF(l \otimes \Uparrow k, \ulcorner Prf(k, \ulcorner \varphi \urcorner) \urcorner).$$

By *FPE*,

$$Prf(l \otimes \Uparrow k, \ulcorner Prf(k, \ulcorner \varphi \urcorner) \urcorner),$$

therefore

$$Prf(c(k), \ulcorner Prf(k, \ulcorner \varphi \urcorner) \urcorner).$$

◀

7.8 Lemma. *The normality conditions for Prf are fulfilled.*

Proof. By *FPE*, *Prf* is provably Δ_1. It follows from *FPE* and 7.6 that for any arithmetical sentence φ

$$\mathcal{PA} \vdash \varphi \text{ if and only if } Prf(n, \ulcorner \varphi \urcorner) \text{ holds for some natural } n.$$

Finiteness of proofs. For each n the set

$$T(k) = \{l \mid Prf(k, l)\}$$

is finite. Indeed, if k is a number of an $\mathcal{LP}$-term, we can use the finiteness of $I(t)$; otherwise we use the normality of *PROOF*. An algorithm for the function from k to the canonical number of $T(k)$ for *Prf* can be constructed from those for *PROOF*, and from the decision algorithm for $I(t)$, Lemma 6.5(1).

Conjoinability of proofs for *Prf* is realized by the function $a(x, y)$ since by Lemma 7.7,

$$T(k) \cup T(n) \subseteq T(a(k, n)).$$

◀

Let us finish the proof of the final "not 1 implies not 5" part of 7.1. Given a sequent $\Gamma \Rightarrow \Delta$ not provable in $\mathcal{LPG}_0^-$ we have constructed an interpretation $*$ such that Γ^* are all true, and Δ^* are all false in the standard model of arithmetic (7.5). Therefore, $(\bigwedge \Gamma \rightarrow \bigvee \Delta)^*$ is false.
◀

7.9 Corollary. *$\mathcal{LP}_0$ is decidable.*

Given an $\mathcal{LP}$-formula F run the saturation algorithm $\mathcal{SA}$ on a sequent $\Rightarrow F$. If $\mathcal{SA}$ fails, then $\mathcal{LP}_0 \vdash F$. Otherwise, $\mathcal{LP}_0 \not\vdash F$.

7.10 Corollary. (Completeness of $\mathcal{LP}$ with respect to the provability semantics.)

$$\begin{array}{lll} \mathcal{LP}(\mathcal{CS}) \vdash F & \Leftrightarrow & \mathcal{PA} \vdash F^* \textit{ for any } \mathcal{CS}\textit{-interpretation } *. \\ & \Leftrightarrow & F^* \textit{ is true for any } \mathcal{CS}\textit{-interpretation } *. \end{array}$$

7.11 Corollary. (Cut elimination in $\mathcal{LP}_0$.) *Every sequent derivable in $\mathcal{LPG}_0$ can be derived without the cut rule.*

Proof. By Theorem 7.1 $\mathcal{LPG}_0^- \vdash \Gamma \Rightarrow \Delta$ iff $\mathcal{LPG}_0 \vdash \Gamma \Rightarrow \Delta$.
◀

7.12 Corollary. (Cut elimination in $\mathcal{LP}$.) *Every sequent derivable in $\mathcal{LPG}$ can be derived without the cut rule.*

Proof. Cut elimination for $\mathcal{LP}$ can be established by a direct system of reductions, and it has been done in [6], [7]. We may also get the cut elimination theorem for $\mathcal{LP}$ as a side product of the arithmetical completeness theorem for $\mathcal{LP}$. Indeed, a straightforward analogue of Theorem 7.1 where $\mathcal{LP}_0$ and $\mathcal{LPG}_0$ are replaced by $\mathcal{LP}$ and $\mathcal{LPG}$ respectively holds. As in 7.1 it suffices to establish that if $\mathcal{LPG} \nvdash \Gamma \Rightarrow \Delta$ then for any constant specification $\mathcal{CS}$ there exists a $\mathcal{CS}$-interpretation $*$ such that the arithmetical sentence $(\bigwedge \Gamma \to \bigvee \Delta)^*$ is false. Let us sketch changes that should be made in the definitions and proofs from Sections 6 and 7 to make them work for $\mathcal{LP}$. Fix a constant specification $\mathcal{CS}$. Definition 6.3 of the saturated sequent should be updated by

7. $\mathcal{CS} \cap \Delta = \emptyset$

The item 7 of the saturation algorithm should be updated by an additional backtracking condition: if $\mathcal{CS} \cap \Delta = \emptyset$ then backtrack. Then Lemma 6.4 holds with the new definition of a saturated sequent and $\mathcal{LPG}^-$ instead of $\mathcal{LPG}_0^-$. Item 3 of Lemma 6.5 should be read as

3. $\mathcal{CS} \in \widetilde{\Gamma}$ and if $t{:}X \in \widetilde{\Gamma} \setminus \mathcal{CS}$, then $X \in \widetilde{\Gamma}$

The new completion algorithm should begin with setting $\Gamma_0 = \{F \mid F \in \Gamma \cup \mathcal{CS}\}$. The rest of 6.5 and the entire 7.1 remain intact under the new definitions.
◀

7.13 Comment. Decidability of $\mathcal{LP}$ follows from the results of [53]. This fact can also be easily obtained from the cut elimination property of $\mathcal{LP}$ (Corollary 7.12).

7.14 Corollary. (Non-emptiness of provability semantics for $\mathcal{LP}$). *For any constant specification $\mathcal{CS}$ there exists a $\mathcal{CS}$- interpretation $*$.*

Proof. An easy inspection of the rules in $\mathcal{LPG}_0$ shows that the sequent $\mathcal{CS} \Rightarrow$ is not derivable in $\mathcal{LPG}_0^-$, and thus $\mathcal{LPG}_0 \nvdash \mathcal{CS} \Rightarrow$. Indeed, if $\mathcal{LPG}_0^- \vdash c{:}\mathbf{A} \Rightarrow$, then $c{:}\mathbf{A}$ is introduced by the rule $(:\ \Rightarrow)$ from a previously derived sequent $\mathbf{A} \Rightarrow$. This is impossible since $\mathbf{A}$ is an axiom of $\mathcal{LP}_0$ and thus $\mathcal{LPG}_0 \vdash\ \Rightarrow \mathbf{A}$: should $\mathcal{LPG}_0 \vdash \mathbf{A} \Rightarrow$, we would have $\mathcal{LPG}_0 \vdash\ \Rightarrow$, which is impossible, e.g. because $\mathcal{LPG}_0^- \nvdash\ \Rightarrow$.

From $\mathcal{LPG}_0 \nvdash \mathcal{CS} \Rightarrow$ it follows that $\mathcal{LPG}_0 \nvdash\ \Rightarrow \neg\mathcal{CS}$. By 7.1, there exists an interpretation $*$ such that $(\neg\mathcal{CS})^*$ is false, i.e. $\mathcal{CS}^*$ is true.
◀

7.15 Comment. The straightforward analogue of Theorem 7.1 holds for the call-by-name semantics (cf. Comment 5.8) as well. Some minor modifications are needed to adapt the proof of 7.1 to this new case. First, we redefine $\mu x.\varphi(x,\vec{y})$ as an arithmetical ι-term

$$\iota z.[\varphi(x,\vec{y}) \wedge \forall z < x \neg\varphi(z,\vec{y})].$$

Then we write down a *Fixed Point Equation* that is similar to *FPE* from 7.1 with some adjustments corresponding to the understanding of $*$ as the call-by-name interpretation, and the new reading of $\mu x.\varphi(x,\vec{y})$ as an arithmetical ι-term (cf.[4], [42],[64]).

7.16 Comment. In [64] a complete axiomatization of the joint logic of proofs with its call-by-name semantics and the formal provability was found. Thus $\mathcal{LP}$ as it was presented in [4] was combined with the logic of formal provability $\mathcal{GL}$ (cf.[12],[14]).

8 Realization of modal and intuitionistic logics

It is easy to see that the forgetful projection of $\mathcal{LP}$ is correct with respect to $\mathcal{S}4$. Let F^o be the result of substituting $\Box X$ for all occurrences of $t\!:\!X$ in F, and $\Gamma^o = \{F^o \mid F \in \Gamma\}$ for any set Γ of $\mathcal{LP}$-formulas.

8.1 Lemma. *If $\mathcal{LP} \vdash F$, then $\mathcal{S}4 \vdash F^o$.*

Proof. This is a straightforward induction on a derivation in $\mathcal{LP}$.
◀

The goal of the current section is to establish the converse, namely that $\mathcal{LP}$ suffices to realize any $\mathcal{S}4$ theorem. By an *$\mathcal{LP}$-realization* of a modal formula F we mean an assignment of proof polynomials to all occurrences of the modality in F. Let F^r be the image of F under a realization r. Positive and negative occurrences of modality in a formula and a sequent are defined in the usual way. Namely

1. an indicated occurrence of $\Box$ in $\Box F$ is positive;

2. any occurrence of $\Box$ from F in $G \to F$, $G \wedge F$, $F \wedge G$, $G \vee F$, $F \vee G$, $\Box F$ and $\Gamma \Rightarrow \Delta, F$ has the same polarity as the corresponding occurrence of $\Box$ in F;

3. any occurrence of $\Box$ from F in $\neg F$, $F \to G$ and $F, \Gamma \Rightarrow \Delta$ has a polarity opposite to that of the corresponding occurrence of $\Box$ in F.

In a provability context $\Box F$ is intuitively understood as "*there exists a proof x of F*". After a skolemization, all negative occurrences of $\Box$ produce arguments of Skolem functions, while positive ones give functions of those arguments. For example, $\Box A \to \Box B$ should be read informally as

$$\exists x \text{ “ } x \textit{ is a proof of } A\text{”} \to \exists y \text{ “ } y \textit{ is a proof of } B\text{”},$$

with the Skolem form

$$\text{“ } x \textit{ is a proof of } A\text{”} \to \text{“ } f(x) \textit{ is a proof of } B\text{”}.$$

The following definition captures this feature: a realization r is *normal* if all negative occurrences of $\Box$ are realized by proof variables.

8.2 Theorem. *If $\mathcal{S}4 \vdash F$, then $\mathcal{LP} \vdash F^r$ for some normal realization r.*

Proof. Consider a cut-free sequent formulation of $\mathcal{S}4$, with sequents $\Gamma \Rightarrow \Delta$, where Γ and Δ are finite multisets of modal formulas. Axioms are sequents of the form $S \Rightarrow S$, where S is a propositional letter, and the sequent $\bot \Rightarrow$. Along with the usual structural rules (weakening, contraction, cut) and rules introducing boolean connectives there are two proper modal rules (cf.[73]):

$$\frac{A, \Gamma \Rightarrow \Delta}{\Box A, \Gamma \Rightarrow \Delta}\,(\Box \Rightarrow) \qquad \text{and} \qquad \frac{\Box\Gamma \Rightarrow A}{\Box\Gamma \Rightarrow \Box A}\,(\Rightarrow \Box)$$

$(\Box\{A_1, \ldots, A_n\} = \{\Box A_1, \ldots, \Box A_n\})$.

If $\mathcal{S}4 \vdash F$, then there exists a cut-free derivation $\mathcal{T}$ of a sequent $\Rightarrow F$. It suffices now to construct a normal realization r such that $\mathcal{LP} \vdash \bigwedge \Gamma^r \to \bigvee \Delta^r$ for any sequent $\Gamma \Rightarrow \Delta$ in $\mathcal{T}$. We will also speak about a sequent $\Gamma \Rightarrow \Delta$ being derivable in $\mathcal{LP}$ meaning $\mathcal{LP} \vdash \bigwedge \Gamma \to \bigvee \Delta$, or, equivalently, $\Gamma \vdash_{\mathcal{LP}} \bigvee \Delta$, or $\mathcal{LPG} \vdash \Gamma \Rightarrow \Delta$. Note that in a cut-free derivation $\mathcal{T}$ the rules respect polarities, all occurrences of $\Box$ introduced by $(\Rightarrow \Box)$ are positive, and all negative occurrences are introduced by $(\Box \Rightarrow)$ or by weakening. Occurrences of $\Box$ are *related* if they occur in related formulas of premises and conclusions of rules; we extend this relationship by transitivity. All occurrences of $\Box$ in $\mathcal{T}$ are naturally split into disjoint *families* of related ones. We call a family *essential* if it contains at least one case of the $(\Rightarrow \Box)$ rule.

Now the desired r will be constructed by steps 1 – 3 described below. We reserve a large enough set of proof variables as *provisional variables*.

Step 1. For every negative family and nonessential positive family we replace all occurrences of $\Box$ by "$x:$" for a fresh proof variable x.

Step 2. Pick an essential family f, enumerate all the occurrences of rules $(\Rightarrow \Box)$ which introduce boxes of this family. Let n_f be the total number of such rules for the family f. Replace all boxes of the family f by the term

$$(v_1 + \ldots + v_{n_f}),$$

where v_i's are fresh provisional variables. The resulting tree $\mathcal{T}_0$ is labelled by $\mathcal{LP}$ formulas, since all occurrences of the kind $\Box X$ in $\mathcal{T}$ are replaced by $t:X$ for the corresponding t.

Step 3. Replace the provisional variables by proof polynomials as follows. Proceed from the leaves of the tree to its root. By induction on the depth of a node in $\mathcal{T}_0$ we establish that after the process passes a node, a sequent assigned to this node becomes derivable in $\mathcal{LP}$. The axioms $S \Rightarrow S$ and $\bot \Rightarrow$ are derivable in $\mathcal{LP}$. For every rule other than $(\Rightarrow \Box)$ we do not change the realization of formulas, and just establish that the concluding sequent is provable in $\mathcal{LP}$ given that the premises are. Moreover, every move down in the tree $\mathcal{T}_0$ other than $(\Rightarrow \Box)$ is a rule of the system $\mathcal{LPG}$, therefore, the induction steps corresponding to these moves follow easily from the equivalence of $\mathcal{LP}$ and $\mathcal{LPG}$.

Let an occurrence of the rule $(\Rightarrow \Box)$ have number i in the numbering of all rules $(\Rightarrow \Box)$ from a given family f. This rule already has a form

$$\frac{y_1:Y_1, \ldots, y_k:Y_k \Rightarrow Y}{y_1:Y_1, \ldots, y_k:Y_k \Rightarrow (u_1 + \ldots + u_{n_f}):Y},$$

where $y_1, \ldots, y_k$ are proof variables, $u_1, \ldots, u_{n_f}$ are proof polynomials, and u_i is a provisional variable. By the induction hypothesis, the premise sequent $y_1:Y_1, \ldots, y_k:Y_k \Rightarrow Y$ is derivable in $\mathcal{LP}$, which yields a derivation of

$$y_1:Y_1, \ldots, y_k:Y_k \Rightarrow Y.$$

By lifting lemma (Proposition 4.4), construct a proof polynomial $t(y_1, \ldots, y_n)$ such that

$$y_1:Y_1, \ldots, y_k:Y_k \Rightarrow t(y_1, \ldots, y_n):Y$$

is derivable in $\mathcal{LP}$. Since

$$\mathcal{LP} \vdash t:Y \to (u_1 + \ldots + u_{i-1} + t + u_{i+1} + \ldots + u_{n_f}):Y$$

we have

$$\mathcal{LP} \vdash y_1:Y_1, \ldots, y_k:Y_k \Rightarrow (u_1 + \ldots + u_{i-1} + t + u_{i+1} + \ldots + u_{n_f}):Y.$$

Now substitute $t(y_1, \ldots, y_n)$ for u_i everywhere in $\mathcal{T}_0$.

By the way, this may lead to the constant specifications of the sort $c{:}\mathbf{A}(c)$ where $\mathbf{A}(c)$ contains c. It looks like such self-referential constant specifications are essential for realization of modal logic in the Logic of Proofs.

Note that $t(y_1,\ldots,y_n)$ has no provisional variables, and that there is one less provisional variable (namely u_i) in the entire tree $\mathcal{T}_0$. All sequents derivable in $\mathcal{LP}$ remain derivable in $\mathcal{LP}$, the conclusion of the given rule $(\Rightarrow\Box)$ became derivable, and the induction step is complete.

Eventually, we substitute terms of non-provisional variables for all provisional variables in $\mathcal{T}_0$ and establish that the corresponding root sequent of $\mathcal{T}_0$ is derivable in $\mathcal{LP}$. Note that the realization of $\Box$'s built by this procedure is normal.

◄

8.3 Corollary. (Arithmetical completeness of $\mathcal{S}4$.) $\mathcal{S}4\vdash F$ *iff there is a realization* r *and a constant specification* $\mathcal{CS}$ *such that* F^r *is* $\mathcal{CS}$*-valid.*

8.4 Comment. It follows from 8.1 and 8.2 that $\mathcal{S}4$ is nothing but a lazy version of $\mathcal{LP}$ that does not distinguish between the proof polynomials. Each theorem of $\mathcal{S}4$ admits a decoding via $\mathcal{LP}$ as a statement about specific proofs. The language of $\mathcal{LP}$ is more rich than that of $\mathcal{S}4$. In particular, $\mathcal{S}4$ theorems admit essentially different realizations in $\mathcal{LP}$. For example, consider two theorems of $\mathcal{LP}$ having the same modal projection:

$$x{:}F\vee y{:}F\rightarrow(x+y){:}F\quad\text{and}\quad x{:}F\vee x{:}F\rightarrow x{:}F.$$

The former of these formulas is a meaningful specification of the operation "+". In a contrast, the latter one is a trivial tautology.

So $\mathcal{LP}$ is the right logic of provability, and $\mathcal{S}4$ should be considered as a lazy higher level language on top of $\mathcal{LP}$. A general recipe for using $\mathcal{S}4$ as a provability logic might be the following: derive in $\mathcal{S}4$ or reason about $\mathcal{S}4$ using a conventional modal logic technique as before, then translate the results into $\mathcal{LP}$ to recover their true provability meaning.

8.5 Comment. As it was noticed by A. Kopylov, the example from 8.4 can be generalized: $\mathcal{S}4$ also admits a degenerated realization in the "+"-free fragment of $\mathcal{LP}$, under which all arguments of proof polynomials are denoted by the same proof variable and only one universal constant is used as a coefficient.

For example, the $\mathcal{S}4$-theorem $(\Box A\vee\Box B)\rightarrow\Box(A\vee B)$ (cf. Example 4.7) can be realized in $\mathcal{LP}$ as $(x:A\vee x:B)\rightarrow(c{\cdot}x):(A\vee B)$ with the constant specification $c:(A\rightarrow A\vee B)$, $c:(B\rightarrow A\vee B)$. As one can see, this realization cripples the provability content of modal logic. Namely, it presupposes that the constant c stands for the proof of two different axioms, which is inconsistent with an injective assignment of proof constants to propositional axioms in rule *R2* of $\mathcal{LP}$. The assumption that A and B have the same proof contradicts the intended provability reading of the original modal formula $(\Box A \vee \Box B) \to \Box(A \vee B)$ as *if there is a proof of A, or there is a proof of B, then there is a proof of $A \vee B$.* Indeed, the Skolem style conversion of this formula from the language with quantifiers into the quantifier-free language with Skolem functions is *if x is a proof of A and y is a proof of B, then $t(x,y)$ is a proof of $A \vee B$.* One can show that such $t(x,y)$ cannot be taken to be "+"-free provided x and y are distinct proof variables. Indeed, let S_1 and S_2 be propositional letters. Suppose

$$\mathcal{LP} \vdash x : S_1 \vee y : S_2 \to t : (S_1 \vee S_2)$$

for some "+"-free term t. Then $\mathcal{LP} \vdash x : S_1 \to t : (S_1 \vee S_2)$ and $\mathcal{LP} \vdash y : S_2 \to t : (S_1 \vee S_2)$. Consider the shortest cut-free derivation $\mathcal{D}$ of $x : S_1 \Rightarrow t : (S_1 \vee S_2)$ in $\mathcal{LPG}$. A straightforward analysis of $\mathcal{D}$ rules out the use of axioms other than $x : S_1 \Rightarrow x : S_1$ and rules other than $(\Rightarrow \cdot)$ and $(\Rightarrow c)$ in the form $x : S_1 \Rightarrow c : \mathbf{A}$. Therefore t is a product of some proof constants and the variable x. Similarly, from $\mathcal{LP} \vdash y : S_2 \to t : (S_1 \vee S_2)$ we conclude that t is a product of some proof constants and the variable y. Therefore, t is a product of some proof constants, and $\mathcal{D}$ does not contain axioms of the sort $x : S_1 \Rightarrow x : S_1$. That means that in the leaf nodes of $\mathcal{D}$ there are only the rules $(\Rightarrow c)$ in the form $x : S_1 \Rightarrow c : \mathbf{A}$. Erase $x : S_1$ from the antecedents of all sequents in $\mathcal{D}$. The remaining tree will be a derivation of $\Rightarrow t : (S_1 \vee S_2)$ in $\mathcal{LPG}$. This would yield $\mathcal{LP} \vdash t : (S_1 \vee S_2)$ and $\mathcal{LP} \vdash S_1 \vee S_2$, which not true.

The "+"-free fragment of $\mathcal{LP}$ is not complete with respect to the class of all single-conclusion proof predicates. It can be made complete by adding the functionality principle from [2]. The completeness of the resulting system $\mathcal{FLP}$ with respect to single-conclusion proof systems was established by V. Krupski in ([42]). $\mathcal{FLP}$ does not have a modal counterpart. For example, $\mathcal{FLP}$ derives a principle $\neg(x : A \wedge x : (A \to A))$, which has the forgetful projection $\neg(\Box A \wedge \Box(A \to A))$. The latter is false in any normal modal logic.

8.6 Definition. Let $gk(F)$ denote a translation of an intuitionistic formula F into the plain modal language that puts the prefix $\Box$ in front of all subformulas in F (*Gödel-Kolmogorov translation*). Under $mt(F)$ we understand the translation that prefixes only atoms and implications in F (*McKinsey-Tarski translation*). A propositional formula F is *GK-realizable* (*MT-realizable*) if there exists a normal realization r such that $gk(F)^r$ ($mt(F)^r$) is derivable in $\mathcal{LP}$.

8.7 Theorem. (Realization of intuitionistic logic) *For any $\mathcal{I}nt$-formula F*

1. $\mathcal{I}nt \vdash F \quad \Leftrightarrow \quad F$ *is GK-realizable,*

2. $\mathcal{I}nt \vdash F \quad \Leftrightarrow \quad F$ *is MT-realizable*

Proof. It is well-known that

$$\mathcal{I}nt \vdash F \;\; \textit{iff} \;\; \mathcal{S}4 \vdash gk(F)$$

(see, for example, [18]), and

$$\mathcal{I}nt \vdash F \;\; \textit{iff} \;\; \mathcal{S}4 \vdash mt(F)$$

([25],[49]). A straightforward combination of these results with the realization of $\mathcal{S}4$ into $\mathcal{LP}$ (Theorem 8.2) brings us the desired result.
◄

8.8 Corollary. (Arithmetical completeness of $\mathcal{I}nt$.) *$\mathcal{I}nt \vdash F$ iff there is a realization r and constant specification $\mathcal{CS}$ such that $gk(F)^r$ is $\mathcal{CS}$-valid ($mt(F)^r$ is $\mathcal{CS}$-valid).*

Note that *GK*-realizability may be regarded as a formalization of the Kolmogorov calculus of problems from [34] by reading "problem solutions" as "proofs". This realizability gives a plausible formalization of Kolmogorov's calculus of problems [34]. Propositional atoms are interpreted as *atomic problems*, namely statements of the sort $t:S$ meaning "*t is a proof of S*". Intuitionistic connectives are given precise meaning according to [34] (cf. the description of *BHK* semantics in section 1).

We conclude this section with examples of *GK*- and *MT*-realizability.

8.9 Example. Let S, T be propositional letters. Consider the formula

$$F \equiv (\neg S \vee T) \rightarrow (S \rightarrow T),$$

obviously provable in $\mathcal{I}nt$. The corresponding translations of this formula to the modal language are (in both cases the outermost $\Box$'s are suppressed for briefty):

$$mt(F) = (\Box\neg\Box S \vee \Box T) \rightarrow \Box(\Box S \rightarrow \Box T),$$

$$gk(F) = \Box(\Box\neg\Box S \vee \Box T) \rightarrow \Box(\Box S \rightarrow \Box T).$$

We will present one of the possible meaningful normal realizations in $\mathcal{LP}$ for each of $mt(F)$ and $gk(F)$.

The following is a derivation in $\mathcal{LP}$ with a simultanious construction of a normal realization of $mt(F)$.

1. $\neg x:S \rightarrow (x:S \rightarrow y:T)$, by classical logic;
2. $a:[\neg x:S \rightarrow (x:S \rightarrow y:T)]$, by necessitation rule 4.5. Note that here a is a product of some axiom constants with obvious specifications;

3. $z:(\neg x:S) \to (a\cdot z):(x:S \to y:T)$, from 2, by *A2*;
4. $y:T \to (x:S \to y:T)$, axiom of propositional logic *A0*;
5. $b:[y:T \to (x:S \to y:T)]$, from 4, by axiom necessitation *R2*;
6. $!y:y:T \to (b\cdot !y):(x:S \to y:T)$, from 5, by *A2*;
7. $y:T \to !y:y:T$, axiom *A3*;
8. $y:T \to (b\cdot !y):(x:S \to y:T)$, from 6,7, by classical logic;
9. $(z:(\neg x:S) \vee y:T) \to (a\cdot z + b\cdot !y):(x:S \to y:T)$, from 3,8, by *A4*.

This realization of $mt(F)$ says: if either z is a proof of $\neg x:S$, or y is a proof of T, then $a\cdot z + b\cdot !y$ is a proof of the implication $x:S \to y:T$, where a and b are proofs of the tautologies $\neg x:S \to (x:S \to y:T)$ and $y:T \to (x:S \to y:T)$ respectively.

In the case of $gk(F)$ the realization is constructed along the following derivation in $\mathcal{LP}$.

1. $\neg x:S \to (x:S \to y:T)$, by classical logic;
2. $z:(\neg x:S) \to \neg x:S$, axiom *A1*;
3. $z:(\neg x:S) \to (x:S \to y:T)$, from 1,2;
4. $y:T \to (x:S \to y:T)$, axiom of propositional logic *A0*;
5. $(z:(\neg x:S) \vee y:T) \to (x:S \to y:T)$, from 3,4, by classical logic;
6. $c:H$, when H is from 5, by necessitation rule 4.5. Here c is a ground proof polynomial, easily recoverable from the derivation of 5.
7. $u:(z:(\neg x:S) \vee y:T) \to (c\cdot u):(x:S \to y:T)$, from 6, by *A2*.

This realization says: if u is a proof of the disjunction $z:\neg x:S \vee y:T$, then $c\cdot u$ is a proof of $x:S \to y:T$, where c is a proof of $(z:\neg x:S \vee y:T) \to (x:S \to y:T)$.

9 Realization of λ-calculi

In the section we show that $\mathcal{LP}$ provides a standard provability semantics for the operator of λ-abstraction. Through a realization in $\mathcal{LP}$ both modality and λ-terms receive a uniform provability semantics.

The defined abstraction operator $\lambda^* x$ on proof polynomials below is a natural extension of the defined λ-abstraction operator $\lambda^* x$ in combinatory logic (cf.[73]).

9.1 Definition. As usual (cf.[73]), the intuitionistic version $\mathcal{ILPG}$ of $\mathcal{LPG}$ may be defined as the fragment of $\mathcal{LPG}$ consisting of sequents of the form $\Gamma \Rightarrow \Delta$, there Δ contains at most one formula.

The cut elimination theorem for $\mathcal{ILPG}$ was established in [6], [7].

9.2 Definition. Under *ground* $(\Rightarrow !)$ rule we mean the rule $(\Rightarrow !)$ where the principal proof polynomial t contains no variables. An $\mathcal{ILPG}$-derivation $\mathcal{D}$ is *pure* if it uses no rules other than $(\Rightarrow \cdot)$, $(\Rightarrow c)$, and ground $(\Rightarrow !)$. It is clear that every pure derivation is normal since it has no cuts.

Assume that a calculus of λ-terms is presented as the sequent calculus of the format $x_1 : A_1, \ldots, x_n : B_n \Rightarrow t(\vec{x}) : B$ with the reading *term* $t(\vec{x})$ *has a type* B *provided* x_i *has type* B_i *for all* $i = 0, 1, \ldots, n$ (cf. system **G2i*** from [73]). Under such formulation a λ-term is presented as a sequent, and formation rules of λ-terms become inference rules in the given sequent calculus.

A straightforward observation shows that some of the λ-terms constructors can be naturally represented as derivation in $\mathcal{ILPG}$. For example, the pairing function introduction rule

$$\frac{\Gamma \Rightarrow t{:}A \qquad \Gamma \Rightarrow s{:}B}{\Gamma \Rightarrow \mathbf{p}(t,s){:}(A \wedge B)}$$

has a natural counterpart $\mathcal{ILPG}$-derivation

$$\cfrac{\cfrac{\cfrac{}{\Gamma \Rightarrow c{:}(A \rightarrow (B \rightarrow (A \wedge B))} \qquad \Gamma \Rightarrow t{:}A}{\Gamma \Rightarrow (c \cdot t){:}(B \rightarrow (A \wedge B))} \qquad \Gamma \Rightarrow s{:}B}{\Gamma \Rightarrow (c \cdot t \cdot s){:}(A \wedge B)}\ .$$

In fact the entire λ-calculus can be embedded into $\mathcal{ILPG}$ ([6], [7]). The key element of this embedding is emulating λ-abstraction in the combinatory logic style (cf.[73]). We define the admissible rule λ^* on sequents in $\mathcal{ILPG}$, which will represent in $\mathcal{ILPG}$ traditional λ-abstraction.

9.3 Theorem. (Definable abstraction) *Let* $\mathcal{D}$ *be a pure* $\mathcal{ILPG}$*-derivation of a sequent*

$$\vec{p}{:}\Gamma, x{:}A \Rightarrow t(x){:}B$$

such that x *does not occur in* $\vec{p}{:}\Gamma, A, B$. *Then one may construct a proof polynomial* $\lambda^* x.t(x)$ *and a pure* $\mathcal{ILPG}$*-derivation* $\mathcal{D}'$ *of the sequent*

$$\vec{p}{:}\Gamma \Rightarrow \lambda^* x.t(x){:}(A \rightarrow B).$$

Proof. The base case is the depth of $\mathcal{D}$ equals one. There are two possibilities.

1. $\mathcal{D}$ is an axiom sequent $\vec{p}{:}\Gamma, x{:}A \Rightarrow t(x){:}B$ and $t(x)$ contains an occurrence of x. Then $t(x) : B = x : A$. Let $\mathcal{D}'$ be the pure derivation of the sequent $\Rightarrow (a \cdot b \cdot c) : (A \rightarrow A)$ where a, b, c are proof constants specified by the conditions (cf.[73], section 1.3.6.)

$a:([A\to((A\to A)\to A)]\to[(A\to(A\to A))\to(A\to A)])$
$b:[A\to((A\to A)\to A)]$
$c:[A\to(A\to A)]$.

Let $\lambda^* x.x = (a\cdot b\cdot c)$. In fact this case coincides with the presentation of $\lambda^* x^A.x$ as $\mathbf{s}^{A,A\to A,A}\mathbf{k}^{A,A\to A}\mathbf{k}^{A,A}$ in combinatory logic (cf.[73]).

2. $\mathcal{D}$ is an axiom sequent $\vec{p}:\Gamma, x:A \Rightarrow t(x):B$ and t does not contain an occurrence of x. Then $t:B\in\vec{p}:\Gamma$ and $\vec{p}:\Gamma\Rightarrow t:B$ is again an axiom sequent. Let $\mathcal{D}'$ be

$$\dfrac{\dfrac{}{\vec{p}:\Gamma\Rightarrow b:(B\to(A\to B))}\,(\Rightarrow c)\qquad \vec{p}:\Gamma\Rightarrow t:B}{\vec{p}:\Gamma\Rightarrow (b\cdot t):(A\to B)}\,(\Rightarrow\cdot)\ .$$

Let $\lambda^* x.t = b\cdot t$. This is the well known equality $\lambda^* x^A.t^B = \mathbf{k}^{B,A}t^B$ of combinatory logic.

The induction step corresponding to the ground $(\Rightarrow !)$ rule is treated similarly to case 2. Consider the case $(\Rightarrow\cdot)$. Let a derivation $\mathcal{D}$ end with

$$\dfrac{\vec{p}:\Gamma, x:A\Rightarrow s:(Y\to B)\qquad \vec{p}:\Gamma, x:A\Rightarrow t:Y}{\vec{p}:\Gamma, x:A\Rightarrow (s\cdot t):B}\ .$$

By the induction hypothesis, we have already built pure derivations of $\vec{p}:\Gamma\Rightarrow\lambda^* x.s:(A\to(Y\to B))$ and $\vec{p}:\Gamma\Rightarrow\lambda^* x.t:(A\to Y)$. From them we construct pure derivations

$$\dfrac{\dfrac{}{\vec{p}:\Gamma\Rightarrow c:((A\to(Y\to B))\to((A\to Y)\to(A\to B)))}\qquad \vec{p}:\Gamma\Rightarrow\lambda^* x.s:(A\to(Y\to B))}{\vec{p}:\Gamma\Rightarrow(c\cdot\lambda^* x.s):((A\to Y)\to(A\to B))}$$

and

$$\dfrac{\vec{p}:\Gamma\Rightarrow(c\cdot\lambda^* x.s):((A\to Y)\to(A\to B))\qquad \vec{p}:\Gamma\Rightarrow\lambda^* x.t:(A\to Y)}{\vec{p}:\Gamma\Rightarrow(c\cdot\lambda^* x.s\cdot\lambda^* x.t):(A\to B)}\ .$$

Let $\lambda^* x.(s\cdot t) = (c\cdot\lambda^* x.s\cdot\lambda^* x.t)$. In combinatory logic notation

$$\lambda^* x^A.s^{Y\to B}t^Y = \mathbf{s}^{A,Y,B}\lambda^* x.s\lambda^* x.t$$

◀

9.4 Comment. In $\mathcal{ILPG}$, λ-abstraction is decoded by a proof polynomials depending on a context (e.g. an $\mathcal{ILPG}$-derivation). In this respect the realization from 9.3 of λ-abstraction by proof polynomials is similar the realization of $\mathcal{S}4$-modality which is decomposed in 8.2 into a set of proof polynomials depending on a context (an $\mathcal{S}4$-derivation).

9.5 Comment. In fact λ^* cannot be easily extended from pure to more general derivations without sacrificing some desired properties. We need to keep the format $\vec{p}{:}\Gamma, x{:}A \Rightarrow t(x){:}B$ throughout the entire derivation $\mathcal{D}$ in order to preserve the inductive character of the definition. The restriction "x does not occur in $\vec{p}{:}\Gamma, A, B$" is needed to guarantee the correctness of β-conversion (below) for λ^*-abstraction, though it rules out $(\Rightarrow !)$. Note that the rule $(\Rightarrow !)$ does not admit abstraction anyway. Indeed, in $\mathcal{ILPG}$ we may derive

$$\frac{x{:}A \Rightarrow x{:}A}{x{:}A \Rightarrow\, !x{:}x{:}A},$$

but for no proof polynomial p does $\mathcal{ILPG}$ derive

$$\Rightarrow p{:}(A \to x{:}A),$$

since $A \to x{:}A$ is not provable in $\mathcal{LP}$.

The dual operation to λ-abstraction i.e. *β-conversion*

$$(\lambda x^A.t^B)s^A \quad \longrightarrow_\beta \quad t^B[x^A/s^A]$$

is naturally presented as the following transformation of pure derivations in $\mathcal{ILPG}$:

$$\frac{\dfrac{\vec{p}{:}\Gamma, x{:}A \Rightarrow t(x){:}B}{\vec{p}{:}\Gamma \Rightarrow \lambda^* x t(x){:}(A \to B)} \qquad \vec{p}{:}\Gamma \Rightarrow s{:}A}{\vec{p}{:}\Gamma \Rightarrow (\lambda^* x t(x)\cdot s){:}B}$$

transforms into

$$\frac{\vec{p}{:}\Gamma \Rightarrow s{:}A \qquad \vec{p}{:}\Gamma, s{:}A \Rightarrow t(s){:}B}{\vec{p}{:}\Gamma \Rightarrow t(s){:}B}.$$

The rule of *η-conversion*

$$(\lambda x^A.t^B)s^A \quad \longrightarrow_\eta \quad t \qquad \textit{if } x \textit{ is not free in } t$$

is treated in the same way. Finally, *α-conversion* corresponds to an obviously valid rule of renaming bounded variables in $\mathcal{ILPG}$-derivations with abstraction.

All other standard λ-term constructors for $\mathcal{I}nt$ can also be realized as admissible rules in $\mathcal{ILPG}$ (cf.[6],[7]). This is a straightforward corollary of the fact that $\mathcal{I}nt$ is a fragment of $\mathcal{ILPG}$

and of the lifting lemma adapted for $\mathcal{ILPG}$. Indeed, if $\mathcal{ILPG} \vdash \Gamma \Rightarrow B$, then by induction on the given proof one can construct a proof polynomial $p(\vec{y})$ such that $\mathcal{ILPG} \vdash \vec{y}\!:\!\Gamma \Rightarrow p(\vec{y})\!:\!B$.

Since both modal logic $\mathcal{S}4$ and all standard λ-term constructors can be emulated by proof polynomials, the Logic of Proofs can also emulate modal λ-calculi. As it was shown in [6], [7] $\mathcal{ILPG}$ naturally realizes the modal λ-calculus for $\mathcal{IS}4$ ([10], [45], [60], cf. also [15]) and thus supplies modal λ-terms with standard provability semantics. This result may be considered as a more general abstract version of the well-known Curry-Howard isomorphism which relates terms/types with proofs/formulas.

10 Discussion

Roughly speaking, $\mathcal{LP}$ is an advanced system of combinatory logic that accommodates not only the "application" operation, but also "proof checker" and "choice". These operations subsume the simply typed λ-calculus together with the modal logic $\mathcal{S}4$, and thus the entire modal λ-calculus. In particular, $\mathcal{LP}$ creates an environment where modality and λ terms are objects of the same nature, namely proof polynomials. Another way to look at it: modal logic is a forgetful projection of a combinatory logic enriched by the operations "proof checker" and "choice".

There was a major difficulty standing in the way of presenting modality via a system of terms: such a presentation should be self-referential and accommodate types containing terms of any type, including its own, for example, $x : F(x)$. The choice of the combinatory logic format for $\mathcal{LP}$ versus the obvious λ-term one in both Gödel's explicit provability logic sketch from [26] and $\mathcal{LP}$ in fact allows a concise presentation of this self-referentiality. The corresponding straightforward λ-term system requires infinite supply of new term constructors and is hardly observable.

The realization of $\mathcal{S}4$ in $\mathcal{LP}$ provides a fresh look at modal logic and its applications in general. Proof polynomials reveal the dynamic character of modality. It raises the general question of finding explicit counterparts to all major modal logics.

Such areas as modal λ-calculi, polymorphic second order λ-calculi, λ-calculi with types depending on terms, non-deterministic λ-calculi, etc., could benefit from viewing their semantics as proof polynomials delivered by $\mathcal{LP}$.

Gabbay's Labelled Deductive Systems ([23]) may serve as a natural framework for $\mathcal{LP}$. Intuitionistic Type Theory by Martin-Löf [46], [47] also makes use of the format $t\!:\!F$ with its informal provability reading. $\mathcal{LP}$ may also be regarded as a basic epistemic logic with explicit justifications; a problem of finding such systems was raised by van Benthem in [9].

The studies of the logic $\mathcal{GL}$ of implicit provability *Provable(x)* ([67],[65],[12], [13],[14],[31]) has given vast experience in arithmetical self-referential semantics for modal logics. The completeness theorem for $\mathcal{LP}$ (Theorem 7.1) could not probably have been obtained without the knowledge accumulated in this area.

11 Acknowledgements

This work has benefited from many interactions over the past several years with H. Barendregt, L. Beklemishev, J. van Benthem, G. Boolos, S. Buss, R. Constable, D. van Dalen, J. Diller, J.M. Dunn, E. Engeler, S. Feferman, D. Gabbay, J.-Y. Girard, W. Hodges, G. Jäger, D. de Jongh, D. Kozen, V. Krupski, S. MacLane, G. Mints, F. Montagna, A. Nerode, E. Nogina, R. Parikh, C. Parsons, W. Pholers, V. Pratt, A. Preller, J. Remmel, D. Roorda, A. Scedrov, R. Shore, T. Sidon, T. Strassen, A. Troelstra, V.A. Uspensky, A. Visser.

Special thanks are due to Robert Constable and Anil Nerode for supporting this research during my work at Cornell University since 1995.

I am indebted to Sam Buss, Volodya Krupski, Bob Milnikel, Lena Nogina, Tanya Sidon, and Fred Smith for a reading different versions of this paper which led to valuable improvements.

The research described in this paper was supported in part by ARO under the MURI program "Integrated Approach to Intelligent Systems", grant DAAH04-96-1-0341, by DARPA under program LPE, project 34145, and by the Russian Foundation for Basic Research, grant 96-01-01395.

References

[1] S. Artemov. "Kolmogorov logic of problems and a provability interpretation of intuitionistic logic", *Theoretical Aspects of Reasoning about Knowledge - III Proceedings*, Morgan Kaufman Pbl., pp. 257-272, 1990

[2] S. Artemov and T. Strassen, "Functionality in the Basic Logic of Proofs", *Tech.Rep. IAM 92-004, Department for Computer Science*, University of Bern, Switzerland, 1993.

[3] S. Artemov, "Logic of Proofs", *Annals of Pure and Applied Logic*, v. 67 (1994), pp. 29-59.

[4] S. Artemov, "Operational Modal Logic," *Tech. Rep. MSI 95-29*, Cornell University, December 1995.

[5] S. Artemov, "Proof realizations of typed λ-calculi," *Tech. Rep. MSI 97-2*, Cornell University, May 1997.

[6] S. Artemov, "Logic of Proofs: a Unified Semantics for Modality and λ-terms," *Tech. Rep. CFIS 98-06*, Cornell University, March 1998.

[7] S. Artemov, "Unified Semantics for Modality and λ-terms via Proof Polynomials," to appear in *Logic, Language and Computation'97, CSLI Publications*, Stanford University, 1998.

[8] J. Avigad and S. Feferman, "Gödel's Functional ("Dialectica") Interpretation". In: *S. Buss, ed., Handbook of Proof Theory*, Elsevier, pp. 337-406, 1998.

[9] J. van Benthem. "Reflections on epistemic logic", *Logique & Analyse*, 133-134, pp. 5-14, 1991

[10] G. Bierman and V. de Paiva, "Intuitionistic necessity revisited", *Proceedings of the Logic at Work Conference,* Amsterdam (December 1992), Second revision, June 1996 (http://theory.doc.ic.ac.uk/tfm/papers.html).

[11] G. Birkhof, "On the structure of abstract algebras", *Proceedings of the Cambridge Philosophical Society*, v.31, pp.433-454, 1935

[12] G. Boolos, *The Unprovability of Consistency: An Essay in Modal Logic*, Cambridge University Press, 1979

[13] G. Boolos, "The logic of provability", *American Mathematical Monthly*, 91, pp.470-480, 1984.

[14] G. Boolos, *The Logic of Provability*, Cambridge University Press, 1993

[15] V. A. J. Borghuis, *Coming to Terms with Modal Logic: On the interpretation of modalities in typed λ-calculus*, Ph.D. Thesis, Technische Universiteit Eindhoven, 1994

[16] L.E.J. Brouwer, "Intuitionistische splitsing van mathematische grondbegrippen" (Dutch), *Nederl. Acad. Wetenssch. Verslagen* 32, 877-880, 1923. German translation *Jahresber. Dtsch. Math.-Ver.* 33, 251-256.

[17] S. Buss, "The Modal Logic of Pure Provability", *Notre Dame Journal of Formal Logic*, v. 31, No. 2, 1990

[18] A. Chagrov and M. Zakharyaschev, *Modal Logic*, Oxford Science Publications, 1997.

[19] R. Constable, "Types in Logic, Mathematics and Programming". In: *S. Buss, ed., Handbook of Proof Theory*, Elsevier, pp. 683-786, 1998.

[20] D. van Dalen, *Logic and Structure*, Springer-Verlag, 1994.

[21] S. Feferman, "A language and axioms for explicit mathematics". In: *J.N. Crossley, ed., Algebra and Logic*, Springer Verlag, pp. 87-139, 1975.

[22] S. Feferman, "Constructive theories of functions and classes". In: *M. Boffa, D. van Dalen, and K. McAloon, eds., Logic Colloquium '78*, North Holland, pp. 159-224, 1979.

[23] D. M. Gabbay, *Labelled Deductive Systems*, Oxford University Press, 1994.

[24] J.-Y. Girard, Y. Lafont, P. Taylor, *Proofs and Types*, Cambridge University Press, 1989.

[25] K. Gödel, "Eine Interpretation des intuitionistischen Aussagenkalkuls", *Ergebnisse Math. Colloq.*, Bd. 4 (1933), S. 39-40.

[26] K. Gödel, "Vortrag bei Zilsel" (1938), in S. Feferman, ed. *Kurt Gödel Collected Works. Volume III*, Oxford University Press, 1995

[27] R. Goldblatt, "Arithmetical necessity, provability and intuitionistic logic", *Theoria*, 44, pp. 38-46, 1978.

[28] A. Heyting, "Die formalen Regeln der intuitionistischen Logik, *Sitzungsberichte der Preussischen Akademie von Wissenschaften. Physikalisch-mathematische Klasse*, pp. 42-56, 1930

[29] A. Heyting, "Die intuitionistische Grundlegung der Matematik", *Erkenntnis* v.2, pp.106-115, 1931.

[30] A.Heyting, *Matematische Grundlagenforschung. Intuitionismus. Beweistheorie*, Springer, Berlin, 1934.

[31] D. de Jongh and G. Japaridze, "Logic of Provability", in S. Buss, ed., *Handbook of Proof Theory*, Elsevier, 1998

[32] S. Kleene. "On the interpretation of intuitionistic number theory", *Journal of Symbolic Logic*, v. 10, pp. 109-124, 1945

[33] S. Kleene. "Classical extensions of intuitionistic mathematics", In *Y. Bar-Hillel, ed. Logic, Methodology and Philosophy of Science 2*, North Holland, pp. 31-44, 1965

[34] A. Kolmogoroff, "Zur Deutung der intuitionistischen Logik", *Math. Ztschr.*, Bd. 35 (1932), S.58-65.

[35] A. Kolmogorov, "About my papers on intuitionistic logic", In: A.N. Kolmogorov, *Selected works*, p. 393, 1985 (Russian), p. 451-452 (English).

[36] D. Kozen and J. Tiuryn, "Logic of Programs", in *Handbook of Theoretical Computer Science. Volume B, Formal Models and Semantics*, The MIT Press/Elsevier, pp. 789-840, 1990

[37] G. Kreisel, "Foundations of intuitionistic logic", in E.Nagel, P.Suppes and A.Tarski, eds., *Logic, Methodology and Philosophy of Science. Proceedings of the 1960 International Congress*, Stanford University Press, Stanford, pp. 198-210, 1962.

[38] G. Kreisel, "On weak completeness of intuitionistic predicate logic", *Journal of Symbolic Logic*, v. 27, pp. 139-158, 1962.

[39] G. Kreisel, "Mathematical Logic", in T.L.Saaty, ed., *Lectures in Modern Mathematics III* Wiley and Sons, New York, pp. 95-195, 1965.

[40] S. Kripke, "Semantical considerations on modal logic", *Acta Philosophica Fennica*, 16, pp. 83-94, 1963.

[41] S. Kripke, "Semantical analysis of intuitionistic logic. I", In: *J.N. Crossley and M.A.E. Dummett, eds., Formal systems and Recursive Functions. Proceedings of the 8th Logic Colloquium* pp. 92-130, North-Holland, 1965.

[42] V.N. Krupski, "Operational Logic of Proofs with Functionality Condition on Proof Predicate", Lecture Notes in Computer Science, v. 1234, *Logical Foundations of Computer Science' 97, Yaroslavl'*, pp. 167-177, 1997

[43] A.V. Kuznetsov and A.Yu. Muravitsky, "The logic of provability", Abstracts of the *4-th All-Union Conference on Mathematical Logic*, p. 73, (Russian), 1976.

[44] E. Lemmon, "New Foundations for Lewis's modal systems", *Journal of Symbolic Logic*, 22, pp. 176-186, 1957.

[45] S. Martini and A. Masini,"A computational interpretation of modal proofs", in Wansing, ed., *Proof Theory of Modal Logics*, (Workshop proceedings), Kluwer, 1994.

[46] P. Martin-Löf. "Constructive mathematics and computer programming", in *Logic, Methodology and Philosophy of Science VI*, North-Holland, pp. 153-175, 1982.

[47] P. Martin-Löf. *Intuitionistic Type Theory*, Studies in Proof Theory, Bibliopolis, Naples, 1984.

[48] J.C.C. McKinsey and A. Tarski, "On closed elements of closure algebras", *Annals of Mathematics*, v. 13, pp. 1-15, 1946.

[49] J.C.C. McKinsey and A. Tarski, "Some theorems about the sentential calculi of Lewis and Heyting", *Journal of Symbolic Logic*, v. 13, pp. 1-15, 1948.

[50] Yu. Medvedev, "Finite problems", *Soviet Mathematics Doklady*, v. 3. pp. 227-230, 1962.

[51] E. Mendelson, *Introduction to mathematical logic. Third edition.*, Wadsworth, 1987.

[52] G. Mints. "Lewis' systems and system T (1965-1973)". In *Selected papers in proof theory*, Bibliopolis, Napoli, 1992.

[53] A. Mkrtychev, "Models for the Logic of Proofs ", Lecture Notes in Computer Science, v. 1234, *Logical Foundations of Computer Science' 97, Yaroslavl'*, pp. 266-275, 1997

[54] R. Montague. "Syntactical treatments of modality with corollaries on reflection principles and finite axiomatizability", *Acta Philosophica Fennica*, 16, pp. 153-168, 1963.

[55] J. Myhill, "Some Remarks on the Notion of Proof", *Journal of Philosophy*, 57, pp. 461-471, 1960

[56] J. Myhill, "Intensional Set Theory", In: S. Shapiro, ed., *Intensional Mathematics*, North-Holland, pp. 47-61, 1985.

[57] I.E. Orlov. "The calculus of compatibility of propositions", *Mathematics of the USSR, Sbornik*, 35, pp.263-286, 1928 (in Russian).

[58] J. van Osten. "A semantical proof of De Jongh's theorem", *Archive for Mathematical Logic*, pp. 105-114, 1991.

[59] C. Parsons and W. Sieg. "Introductory note to *1938a". In: S. Feferman, ed. *Kurt Gödel Collected Works. Volume III*, Oxford University Press, pp. 62-85, 1995.

[60] F. Pfenning and H.C. Wong, "On a modal lambda-calculus for S4", *Electronic Notes in Computer Science* 1, 1995.

[61] H. Rasiowa and R. Sikorski, *The Mathematics of Metamathematics*, Polish Scientific Publishers, 1963.

[62] S. Shapiro. "Intensional Mathematics and Constructive Mathematics". In: S. Shapiro, ed., *Intensional Mathematics*, North-Holland, pp. 1-10, 1985.

[63] S. Shapiro. "Epistemic and Intuitionistic Arithmetic". In: *S. Shapiro, ed., "Intensional mathematics"*, North-Holland, pp. 11-46, 1985.

[64] T. Sidon, "Provability Logic with Operations on Proofs", Lecture Notes in Computer Science, v. 1234, *Logical Foundations of Computer Science' 97, Yaroslavl'*, pp. 342-353, 1997

[65] C. Smorynski, *Self-Reference and Modal Logic*, Springer-Verlag, Berlin, 1985

[66] R. Smullyan, *Diagonalization and Self-Reference*, Oxford University Press, 1994

[67] R. Solovay, "Provability interpretations of modal logic", *Israel Journal of Mathematics*, 25, pp. 287-304, 1976.

[68] G. Takeuti, *Proof Theory*, North-Holland, 1975

[69] A.S. Troelstra, "The scientific work of A. Heyting", In: *D. van Dalen, et al. Logic and Foundations of Mathematics*, Wolters-Noordhoff Publishing, 1968.

[70] A.S. Troelstra "Introductory note to [25]". In: S. Feferman, ed. *Kurt Gödel Collected Works. Volume I*, Oxford University Press, pp. 296-299, 1986.

[71] A.S. Troelstra "Realizability". In *S. Buss, ed., Handbook of Proof Theory*, Elsevier, pp. 407-474, 1998.

[72] A.S. Troelstra and D. van Dalen, *Constructivism in Mathematics. An Introduction*, v. 1, Amsterdam; North Holland, 1988.

[73] A.S. Troelstra and H. Schwichtenberg, *Basic Proof Theory*, Cambridge University Press, 1996.

[74] V.A. Uspensky, "Kolmogorov and mathematical logic", *Journal of Symbolic Logic*, 57, No.2, 1992.

[75] V.A. Uspensky and V.E. Plisko "Intuitionistic Logic", In S.M. Nikol'ski, ed. *A.N. Kolmogorov, Collected works. Mathematics and Mechanics*, pp. 394-404, 1985 (in Russian).

[76] S. Weinstein, "The intended interpretation of intuitionistic logic", *Journal of Philosophical Logic*, 12, pp. 261-270 1983.