THE BULLETIN OF SYMBOLIC LOGIC
Volume 7, Number 1, March 2001

EXPLICIT PROVABILITY AND CONSTRUCTIVE SEMANTICS

SERGEI N. ARTEMOV

Abstract. In 1933 Gödel introduced a calculus of provability (also known as modal logic **S4**) and left open the question of its exact intended semantics. In this paper we give a solution to this problem. We find the logic **LP** of propositions and proofs and show that Gödel's provability calculus is nothing but the forgetful projection of **LP**. This also achieves Gödel's objective of defining intuitionistic propositional logic **Int** via classical proofs and provides a Brouwer-Heyting-Kolmogorov style provability semantics for **Int** which resisted formalization since the early 1930s. **LP** may be regarded as a unified underlying structure for intuitionistic, modal logics, typed combinatory logic and λ-calculus.

§1. A need for a theory of proofs. According to Brouwer, intuitionistic truth means provability. Here is a summary from *Constructivism in Mathematics* by Troelstra and van Dalen ([105], p. 4):

> "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."

In 1931–34 Heyting and Kolmogorov made Brouwer's definition of intuitionistic truth explicit, though informal, by introducing what is now known as the *Brouwer-Heyting-Kolmogorov* (*BHK*) *semantics* ([53], [54], [58]). The *BHK* semantics is widely recognized as the intended semantics for intuitionistic logic ([33], [34], [43], [62], [74], [78], [104], [105], [107], [108], [111], [114]). Its description uses the unexplained primitive notions of *construction* and *proof* (Kolmogorov used the term *problem solution* for the latter). It stipulates that

- a proof of $A \wedge B$ 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[1],

Received September 17, 1998; revised October 10, 2000.

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.

[1]Neither Heyting's paper [54] nor Kolmogorov's [58] contains the well-known extra condition on the disjunction: *a proof of a disjunction should also specify which one of the disjuncts*

1079-8986/01/0701-0001/$4.60

- a proof of $A \to B$ is a construction which, given a proof of A, returns a proof of B,
- absurdity $\bot$ is a proposition which has no proof, a proof of $\neg A$ is a construction which, given a proof of A, would return a proof of $\bot$.

The significance of formalizing the *BHK* semantics extends far beyond justifying the particular choice of axioms for constructive (intuitionistic) logic made by Heyting in 1930 [52]. Provability and proofs as objects appear in many other areas of logic and applications such as modal logics and logics of knowledge, λ-calculus and typed theories, nonmonotonic reasoning, automated deduction and formal verification. Logical systems with built-in provability were anticipated by Kolmogorov and Gödel since the 1930s. Kolmogorov in ([59]) commented on his *BHK* paper of 1932:

> "The paper [58] was written with the hope that the logic of solutions of problems would later become a regular part of courses on logic. It was intended to construct a unified logical apparatus dealing with objects of two types—propositions and problems."

The traditional mathematical model based on the arithmetical predicate of formal provability "there exists a code of a proof of F" (§§3–4), fell short of meeting the above expectations. The unsolved *BHK* problem, as well as a closely related open problem of the intended semantics for Gödel's provability calculus **S4**, indicated that the basic theory of provability had not been built yet.

In this paper we introduce the Logic of Proofs **LP** dealing with both propositions and proofs. This logic is supplied with the adequate provability semantics, decidability and normalization theorems. We give solutions to the problem of the intended semantics for Gödel's provability calculus **S4** and to the problem of *BHK* semantics for intuitionistic propositional logic **Int** (also known as **IPC**) along the lines of Gödel's papers [44], [46]. These and other applications suggest that **LP** fills a certain gap in the foundations of proof theory.

§2. Semantics for intuitionistic logic.

Semantics of intuitionism is a well-developed area. This paper does not intend to teach "the true semantics" for **Int** but rather to show that **LP** achieves Kolmogorov and Gödel's objective to define **Int** via classical proofs.

There is an important distinction between Heyting's and Kolmogorov's descriptions of the *BHK* semantics. Presumably, Heyting intended to explain intuitionistic logic via special intuitionistic understanding of *construction*

---

it is a proof of. This condition is redundant for the usual notion of proof since the predicate "p is a proof of F" is decidable: given a proof p we always know which one of the disjuncts it is a proof of. However, a similar condition appeared in Kleene realizability (cf. [56]), where it makes perfect sense, since the predicate "r realizes F" is undecidable.

and *proof*. Since Heyting's formalization of the axiom system for intuitionistic logic ([52]), an impressive array of intuitionistically justified semantics for intuitionistic logic have been introduced by Kreisel, Kripke, Dyson, van Dalen, Leivant, Veldman, de Swart, Dummet, Troelstra, H. Friedman, Visser, and others (cf. [110]). Those studies lie outside the scope of this paper. We note however that they do not seem to produce a satisfactory formalization of the *BHK* semantics. Here is a summary from van Dalen's chapter "Intuitionistic Logic" in *Handbook of Philosophical Logic*, v. 3 (1986), p. 243 ([110]):

> "The intended interpretation of intuitionistic logic as presented by Heyting ... so far has proved to be rather elusive. ... However, ever since Heyting's formalization, various, more or less artificial, semantics have been proposed."

Kolmogorov in 1932 (and then Gödel in 1933) intended to interpret **Int** on the basis of the usual mathematical notion of proof (problem solution), and thus to provide a *definition* of **Int** within classical mathematics independent of intuitionistic assumptions. In this paper we follow the Kolmogorov-Gödel approach. In agreement with Gödel's papers [44] and [46] we assume that *BHK* "proofs" should correspond to proofs in some formal theory T

$$T \vdash F \iff \textit{there is a BHK proof of } F,$$

and the predicate "p is a *BHK* proof of F" should be decidable. Naturally, we also expect a *BHK* semantics to be non-circular. In particular, *BHK* proofs should not denote derivations in **Int** itself.

Here is the list of major known classical semantics for intuitionistic logic[2].

1. Algebraic semantics (Birkhoff, 1935, [26])
2. Topological semantics (Stone, 1937; Tarski, 1938, [76])
3. Realizability semantics (Kleene, 1945, [56])
4. Beth models (1956, [24])
5. Dialectica Interpretation (Gödel, 1958, [45])
6. Curry-Howard isomorphism (1958, [35])
7. Medvedev's logic of problems (1962, [78])
8. Kripke models (1965, [66])
9. Kuznetsov-Muravitsky-Goldblatt interpretation (1976, [47], [70])
10. Categorical semantics (Goldblatt, 1979, [48])

Those interpretations have shown to be very instrumental for understanding intuitionistic logic though none of them qualifies as a *BHK* semantics.

Interpretations 1–5, 7, 8, 10 are not related to provability. In particular, Kleene realizability disclosed a fundamental *computational* content of formal intuitionistic derivations which is however totally different from the provability semantics. Kleene realizers are not proofs in a formal theory,

[2]Comprehensive surveys of these and other semantics for intuitionistic logic can be found in [33], [92], [105].

the predicate "r realizes F" is not decidable. Kleene himself denied any connection of his realizability with *BHK* interpretation. It is also worth mentioning that Kleene realizability is not adequate for **Int**, i.e., there are realizable propositional formulas not derivable in **Int** (cf. [33], p. 53).

The Curry-Howard isomorphism transliterates natural derivations in **Int** to typed λ-terms. It is a very important generic *functional* reading of logical derivations. However, as *proof objects* Curry-Howard λ-terms denote nothing but derivations in **Int** itself and thus yield a circular provability semantics for the latter. Note that the Curry-Howard isomorphism does not specify **Int**; decent λ-calculi can be built for a variety of logics including proper fragments of **Int**, classical logic, etc. ([21], [88], [89]).

Abstract computational and functional semantics for **Int** which did not address the issue of the original *BHK* semantics for **Int** were also studied in [71], [94] and many other papers (cf. [18], [22], [106]).

Kuznetsov-Muravitsky-Goldblatt semantics for **Int** is based on a non-constructive notion "classically true and formally provable" incompatible with the *BHK* semantics. In particular, it does not contain any *BHK constructions* or *proofs* whatsoever. As far as **S4** is concerned the Kuznetsov-Muravitsky-Goldblatt semantics turned out not to be adequate ([27], [29]).

An attempt to formalize the *BHK* semantics directly was made by Kreisel in his theory of constructions ([62], [64]). The original variant of the theory was inconsistent, and difficulties there occurred already at the propositional level. Goodman ([51]) in 1970 fixed that gap but his solution involved a stratification of constructions into levels which ruined the *BHK* character of this semantics. In particular, a proof of $A \to B$ was no longer a construction that could be applied to any proof of A. A comprehensive account of the Kreisel-Goodman theory could be found in S. Weinstein's paper [114] of 1983, which concludes that

> "The interpretation of intuitionistic theories in terms of the notions of proof and construction ... has yet, however, failed to receive a definitive formulation."

§3. Defining intuitionistic logic in classical provability logic. One of the first papers on provability semantics for intuitionistic logic was published in 1928 by Orlov ([87]) who suggested prefixing all subformulas of a formula by a provability operator. Gödel in 1933 ([44]) introduced the modal calculus of provability (essentially equivalent to the Lewis modal system **S4**) and defined **Int** in this logic. Gödel's provability calculus is based on the classical propositional logic and has the modal axioms and rules

$$\Box F \to F,$$
$$\Box(F \to G) \to (\Box F \to \Box G),$$
$$\Box F \to \Box\Box F,$$
$$F \vdash \Box F \quad (\textit{necessitation} \text{ rule}).$$

Gödel considered the translation $t(F)$ of an intuitionistic formula F into the classical modal language: "box each subformula of F" apparently regarding such a translation to be a fair formalization of the Brouwer thesis

$$\textit{intuitionistic truth} = \textit{provability}.$$

Gödel established that

$$\mathbf{Int} \vdash F \Longrightarrow \mathbf{S4} \vdash t(F),$$

thus providing a reading of **Int**-formulas as statements about classical provability. He conjectured that the converse ($\Longleftarrow$) also held and concluded in [46], pp. 100–101: *Intuitionismus ist daraus ableitbar*[3]. The ($\Longleftarrow$) conjecture was proved in 1948 by McKinsey and Tarski ([77]). The ultimate goal, however, of defining **Int** via classical proofs had not been achieved, because **S4** was left without an exact intended semantics of the provability operator $\Box$:

$$\mathbf{Int} \hookrightarrow \mathbf{S4} \hookrightarrow \ldots ? \ldots \hookrightarrow \textit{REAL PROOFS}$$

It is clear from [44] and [46] that by *REAL PROOFS* Gödel meant systems based on a proof predicate $\mathrm{Proof}(x, y)$ denoting "x is the code of a proof of the formula having a code y" for a classical first order theory containing Peano arithmetic **PA**. Gödel in [44] identified a problem there and pointed out that the straightforward reading of $\Box F$ as the formal provability predicate $\mathrm{Provable}(F) = \exists x \ \mathrm{Proof}(x, F)$ did not work.

> Let $\bot$ be the boolean constant **false** and $\Box F$ be $\mathrm{Provable}(F)$. Then $\Box\bot \rightarrow \bot$ corresponds to the statement *Consis* **PA** expressing consistency of **PA**. An **S4**-theorem $\Box(\Box\bot \rightarrow \bot)$ expresses the assertion that *Consis* **PA** is provable in **PA**, which is false according to the second Gödel incompleteness theorem.

Gödel's paper [44] left open two natural problems concerning

1. the exact intended semantics of Gödel's provability calculus **S4**,
2. the modal logic of the formal provability predicate $\mathrm{Provable}(F)$.

It was already clear, however, that 1 and 2 led to essentially different models of Provability, each targeting its own set of applications (cf. §12).

Problem 2 was solved by R. Solovay [100] who showed that the modal logic **L**[4] axiomatized all propositional properties of the formal provability, and by Artemov [4] and Vardanyan [112] who demonstrated that the first order logic of formal provability was not axiomatizable.

Problem 1 receives a solution in this paper (cf. also technical reports [7], [9]).

The issue of provability semantics for **S4** was addressed by Lemmon [72], Myhill [84], [85], Kripke [65], Montague [83], Novikov [86], Mints [80],

[3] *Intuitionism is derivable from this.*

[4] So called Löb's logic, also known under the names **G**, **GL**, **K4.W**, **PRL**.

Kuznetsov and Muravitsky [70], Goldblatt [47], Boolos [27], [29], Shapiro [95], [96], Buss [32], Artemov [5], and many others. However, there were no adequate Gödelian provability semantics for **S4** found[5]. Moreover, in [83] the problem was announced hopeless.

§**4. Explicit vs. implicit approaches.** The above difficulties with reading **S4**-modality $\Box F$ as $\exists x\ \mathrm{Proof}(x, y)$ are caused by the non-constructive character of the existential quantifier. In particular, in a given model of arithmetic an element that instantiates the existential quantifier over proofs may be nonstandard. In that case $\exists x\ \mathrm{Proof}(x, F)$ though true in the model, does not deliver a "real" **PA**-derivation and thus the *reflection principle*

$$\mathrm{Provable}(F) \rightarrow F$$

is not internally provable. On the other hand, the *explicit reflection principle*

$$\mathrm{Proof}(n, F) \rightarrow F$$

is internally provable for each natural number n. Indeed, if $\mathrm{Proof}(n, F)$ holds, then F is provable. If $\mathrm{Proof}(n, F)$ does not hold then its negation $\neg\,\mathrm{Proof}(n, F)$ is provable, since $\mathrm{Proof}(x, y)$ is a decidable relation. In both cases $\mathrm{Proof}(n, F) \rightarrow F$ is provable.

This consideration suggests the idea of developing an explicit provability logic by switching from the formal provability $\exists x\ \mathrm{Proof}(x, F)$ to $\mathrm{Proof}(t, F)$ and replacing the existential quantifiers on proofs by Skolem style operations on proofs.

Some of those operations appeared in the proof of Gödel's second incompleteness theorem. Within that proof (cf. [27], [29], [79], [98]), in order to prove what are now known as Hilbert-Bernays-Löb derivability conditions, one constructs computable functions $\mathbf{m}(x, y)$ and $\mathbf{c}(x)$ such that

$$\mathbf{PA} \vdash \mathrm{Proof}(s, F \rightarrow G) \wedge \mathrm{Proof}(t, F) \rightarrow \mathrm{Proof}(\mathbf{m}(s, t), G);$$

$$\mathbf{PA} \vdash \mathrm{Proof}(t, F) \rightarrow \mathrm{Proof}(\mathbf{c}(t), \mathrm{Proof}(t, F)).$$

Then these facts are usually relaxed to their simplified versions

$$\mathbf{PA} \vdash \mathrm{Provable}(F \rightarrow G) \wedge \mathrm{Provable}(F) \rightarrow \mathrm{Provable}(G),$$

$$\mathbf{PA} \vdash \mathrm{Provable}(F) \rightarrow \mathrm{Provable}(\mathrm{Provable}(F)),$$

sufficient to establish the incompleteness theorem.

In one of his lectures in 1938, first published in 1995 ([46], cf. [90]), Gödel once again acknowledged the problem of provability semantics for **S4** and mentioned a possibility of building an explicit version of **S4** with basic propositions "t is a proof of F" ($t : F$ in our notation) in order to get a semantics of proofs for **Int**. Though neither definitions nor axiomatization

[5]There are many adequate non-provability models for **S4** known: algebraic, topological, Kripke, etc. (cf. [33], [61]).

were given, Gödel's suggestion specified the format $t\!:\!F$ of an expected solution of the provability semantics problem for **S4** and for the *BHK* problem[6]. It turned out that in addition to Gödelian operations $\mathbf{m}(x, y)$ and $\mathbf{c}(x)$ one more operation on proofs is needed to capture the whole of **S4**.

The particular goals of this paper are

1. *To find a complete axiom system for a classical propositional logic with additional atoms "t is a proof of F" sketched by Gödel in* ([46]) *and to give an intended semantics for Gödel's provability calculus* **S4**.

We introduce a system of computable operations on proofs (*proof polynomials*) and establish the soundness and completeness of the resulting *Logic of Proofs* **LP** (Theorem 8.1 and Corollary 8.10). We show that **LP** realizes all of **S4** (Theorem 9.4 and Corollary 9.5). This gives an adequate provability model for **S4** along the lines of Gödel's suggestions (Corollary 9.6).

2. *To formalize the classical BHK semantics for* **Int** *and to establish the completeness of intuitionistic logic with respect to this semantics.*

We consider realizations of **Int** by proof polynomials based on the Gödel embedding of **Int** in **S4**, and establish that this semantics is adequate (Theorem 9.9). This confirms Kolmogorov's assumption of 1932 that intuitionistic logic **Int** is the calculus of proofs (solutions to problems) in classical mathematics ([58], [59]) and achieves the original objective by Gödel ([44]) to define **Int** via the classical notion of proof.

3. *To enhance typed combinatory logic and typed λ-calculus.*

We show that **LP** is an advanced system of typed combinatory logic and typed λ-calculus with iterated and multiple type assignments. **LP** admits types depending on terms, e.g., of the form $s : (t : F)$, and allows terms to have any given finite set of types.

Through realizations in **LP** both modality and λ-terms may be regarded as systems of proof polynomials.

§5. Logic of Proofs.

Definition 5.1. The language of Logic of Proofs (**LP**) contains

- the language of classical propositional logic which includes propositional variables, truth constants $\top$, $\bot$, and boolean connectives
- proof variables $x_0, \dots, x_n, \dots$, proof constants $a_0, \dots, a_n, \dots$
- function symbols: monadic !, binary $\cdot$ and $+$

[6]We began working on logics with the atoms "t is a proof of F" in 1992, first in collaboration with Gerhard Jäger and Tyko Strassen. The author discovered the Logic of Proofs **LP** during an extended visit to the University of Amsterdam in 1994 before Gödel's lecture [46] was published. An early version of **LP** was presented at logic seminars in Amsterdam and Münster in the fall of 1994.

- operator symbol of the type "*term*: *formula*".

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

$$t ::= x \mid a \mid !t \mid t_1 \cdot t_2 \mid t_1 + t_2.$$

We call these terms *proof polynomials* and denote them by $p, r, s, \ldots$. Constants correspond to proofs of a finite fixed set of axiom schemas. We will omit "$\cdot$" whenever it is safe. We also assume that $p \cdot r \cdot s \ldots$ should be read as $(\ldots((p \cdot r) \cdot s) \ldots)$, and $p + r + s \ldots$ as $(\ldots((p + r) + s) \ldots)$.

Using t to stand for any term and S for any propositional letter, $\top$ or $\bot$, formulas are defined by the grammar

$$F ::= S \mid F_1 \rightarrow F_2 \mid F_1 \wedge F_2 \mid F_1 \vee F_2 \mid \neg F \mid t : F.$$

We will use A, B, C, F, G, H for the formulas in this language, and Γ, $\Delta, \ldots$ for the finite sets (also finite multisets, or finite lists) of formulas unless otherwise explicitly stated. We will also use $\vec{x}, \vec{y}, \vec{z}, \ldots$ and $\vec{p}, \vec{r}, \vec{s}, \ldots$ for vectors of proof variables and proof polynomials respectively. If $\vec{s} = (s_1, \ldots, s_n)$ and $\Gamma = (F_1, \ldots, F_n)$, then $\vec{s} : \Gamma$ denotes $(s_1 : F_1, \ldots, s_n : F_n)$, $\bigvee \Gamma = F_1 \vee \cdots \vee F_n$, $\bigwedge \Gamma = F_1 \wedge \cdots \wedge F_n$. 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, $|s|$ is the length of s, i.e., the total number of symbols in s. We will skip the Gödel number symbol "$\ulcorner\ \urcorner$" inside proof formulas and provability formulas (such as Prf, Proof, Provable, etc.) when it is safe.

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 semantics for $p : F$ are *multi-conclusion* ones, i.e., p may be a proof of several different F's (see Comment 5.7).

Definition 5.2. We define the system $\mathbf{LP}_0$ in the language of **LP**. Axioms:

A0. *Finite set of axiom schemes of classical propositional logic*
A1. $t : F \rightarrow F$ "*reflection*"
A2. $t : (F \rightarrow G) \rightarrow (s : F \rightarrow (t \cdot s) : G)$ "*application*"
A3. $t : F \rightarrow !t : (t : F)$ "*proof checker*"
A4. $s : F \rightarrow (s + t) : F,\ t : F \rightarrow (s + t) : F$ "*sum*"

Rule of inference:

R1. $F \rightarrow G,\ F \vdash G$ "*modus ponens*".

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

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

A *Constant Specification* ($\mathcal{CS}$) is a finite set of formulas $c_1 : A_1, \ldots, c_n : A_n$ such that c_i is a constant, and A_i an axiom A0–A4. $\mathcal{CS}$ is *injective* if for each constant c there is at most one formula $c : A \in \mathcal{CS}$ (each constant denotes a proof of not more than one axiom). Each derivation in **LP** naturally generates the $\mathcal{CS}$ consisting of all formulas introduced in this derivation by the *axiom necessitation* rule. For a constant specification $\mathcal{CS}$, by $\mathbf{LP}(\mathcal{CS})$ we mean $\mathbf{LP}_0$ plus formulas from $\mathcal{CS}$ as additional axioms.

COMMENT 5.3. Atomic constant terms (*combinators*) of typed combinatory logic (cf. [104]) may be regarded as proof constants. The combinator $\mathbf{k}^{A,B}$ of the type $A \to (B \to A)$ can be identified with a constant a specified as $a : (A \to (B \to A))$. The combinator $\mathbf{s}^{A,B,C}$ of the type $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$ corresponds to a constant b such that $b : [(A \to (B \to C)) \to ((A \to B) \to (A \to C))]$. Term variables of combinatory logic may be regarded as proof variables in **LP**, application as operation "$\cdot$". In general a combinatory term t of the type F is represented by an **LP**-formula $t : F$. Typed combinatory logic $\mathbf{CL}_{\to}$ thus corresponds to a fragment of **LP** consisting 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 us 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 **LP** to injective constant specifications only without changing the ability of **LP** to emulate modal logic, or the functional and arithmetical completeness theorems for **LP** (below). However, we choose not to insist on injective constant specifications a priori.

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

$$\Gamma, A \vdash B \implies \Gamma \vdash A \to B,$$

and the substitution lemma: *If* $\Gamma(x, P) \vdash B(x, P)$, *then for any* t, F

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

Obviously,

$$\begin{array}{l} F \text{ is derivable in } \mathbf{LP} \text{ with a} \\ \text{constant specification } \mathcal{CS} \end{array} \iff \mathbf{LP}(\mathcal{CS}) \vdash F \iff \mathbf{LP}_0 \vdash \bigwedge \mathcal{CS} \to F.$$

LEMMA 5.4 (Lifting Lemma). *If* $\vec{s} : \Gamma,\ \ \Delta \vdash_{\mathbf{LP}} F$, *then there is a proof polynomial* $t(\vec{x}, \vec{y})$ *such that*

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

Moreover, if the constant specification $\mathcal{CS}$ *in the original derivation is injective then the resulting constant specification is also injective and extends* $\mathcal{CS}$.

PROOF. By induction on the derivation $\vec{s}:\Gamma,\ \Delta \vdash F$. If $F = s:G \in \vec{s}:\Gamma$, then put $t := !s$ 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 derived 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 from $\vec{s}:\Gamma, \vec{y}:\Delta$. By A2, $\vec{s}:\Gamma, \vec{y}:\Delta \vdash (u \cdot v):F$, and we put $t := u \cdot v$. If F is derived by R2, then $F = c:A$ for some axiom A. Use the same R2 followed by A3: $c:A \to !c:c:A$ and *modus ponens* to get $!c:F$, and put $t := !c$. ⊣

It is easy to see from the proof that the lifting polynomial t is nothing but a blueprint of a given derivation of F. Thus **LP** internalizes its own proofs as proof terms.

COROLLARY 5.5 (Necessitation rule for **LP**).

$$\vdash F \Longrightarrow \vdash p:F \text{ for some ground proof polynomial } p.$$

We shall see in §9 that **LP** suffices to emulate all **S4**-derivations. Meanwhile Example 5.6 below shows how to derive in **LP** vs. **S4**.

EXAMPLE 5.6. We first derive $\Box A \vee \Box B \to \Box(\Box A \vee \Box B)$ in **S4**.

1. $\Box A \to \Box A \vee \Box B,\ \Box B \to \Box A \vee \Box B$, axioms
2. $\Box(\Box A \to \Box A \vee \Box B),\ \Box(\Box B \to \Box A \vee \Box B)$, by necessitation, from 1
3. $\Box A \to \Box\Box A,\ \Box B \to \Box\Box B$, axioms
4. $\Box\Box A \to \Box(\Box A \vee \Box B),\ \Box\Box B \to \Box(\Box A \vee \Box B)$, from 2
5. $\Box A \to \Box(\Box A \vee \Box B),\ \Box B \to \Box(\Box A \vee \Box B)$, from 3, 4
6. $\Box A \vee \Box B \to \Box(\Box A \vee \Box B)$, from 5.

And here is the corresponding derivation in **LP**:

1. $x:A \to x:A \vee y:B,\ y:B \to x:A \vee y:B$, axioms
2. $a:(x:A \to x:A \vee y:B),\ b:(y:B \to x:A \vee y:B)$, constant specification
3. $x:A \to !x:x:A,\ y:B \to !y:y:B$, axioms
4. $!x:x:A \to (a\cdot !x):(x:A \vee y:B),\ !y:y:B \to (b\cdot !y):(x:A \vee y:B)$, from 2
5. $x:A \to (a\cdot !x):(x:A \vee y:B),\ y:B \to (b\cdot !y):(x:A \vee y:B)$, from 3, 4

$5'$. $(a\cdot !x):(x:A \vee y:B) \to (a\cdot !x + b\cdot !y):(x:A \vee y:B)$, an axiom
$5''$. $(b\cdot !y):(x:A \vee y:B) \to (a\cdot !x + b\cdot !y):(x:A \vee y:B)$, an axiom
6. $x:A \to (a\cdot !x + b\cdot !y):(x:A \vee y:B)$, from 5, $5'$
$6'$. $y:B \to (a\cdot !x + b\cdot !y):(x:A \vee y:B)$, from 5, $5''$
$6''$. $x:A \vee y:B \to (a\cdot !x + b\cdot !y):(x:A \vee y:B)$, from 6, $6'$.

COMMENT 5.7. The operations "$\cdot$" and "!" are present in single-conclusion as well as in multi-conclusion proof systems. On the other hand, "+" is an operation for multi-conclusion proof systems only. Indeed, by *A4* we have $s:F \wedge t:G \to (s+t):F \wedge (s+t):G$, thus $s+t$ proves both F and G. 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 [6], [16], and in [67], [68] where it received a complete axiomatization (system **FLP**). However, **FLP** is not compatible with any modal logic. For example, **FLP** derives $\neg(x{:}\top \wedge x{:}(\top\wedge\top))$, which has the forgetful projection $\neg(\Box\top\wedge\Box(\top\wedge\top))$. The latter is false in any normal modal logic. Therefore, provability as a modal operator corresponds to multi-conclusion proof systems.

Single operators "$t{:}$ " in **LP** are not normal modalities since they do not satisfy the property $t{:}(P\rightarrow Q)\rightarrow(t{:}P\rightarrow t{:}Q)$. This makes **LP** essentially different from polymodal logics, e.g. the dynamic logic of programs ([60]), where the modality is upgraded by some additional features. Rather in the Logic of Proofs the modality has been decomposed into a family of proof polynomials (see §9).

§6. Standard provability interpretation of LP. In §6 and §8 by Δ_1 and Σ_1 we mean the corresponding classes of arithmetical predicates. We will use x, y, z to denote individual variables in arithmetic and hope that the reader is able to distinguish them from the proof variables. If n is a natural number, then $\overline{n}$ will denote the numeral corresponding to n, i.e., the standard arithmetical term $0'''^{\cdots}$ where $'$ is the 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.

Definition 6.1. We assume that the first order Peano Arithmetic **PA** contains terms for all primitive recursive functions (cf. [98], [101]), 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 **PA** to a standard Σ_1-formula; φ is *provably* Δ_1 if and only if both φ and $\neg\varphi$ are provably Σ_1.

Definition 6.2. A *proof predicate* is a provably Δ_1-formula $\mathrm{Prf}(x,y)$ such that for every arithmetical sentence φ

$$\mathbf{PA}\vdash\varphi \iff \text{for some } n\in\omega \iff \mathrm{Prf}(n,\ulcorner\varphi\urcorner) \text{ holds.}$$

$\mathrm{Prf}(x,y)$ is *normal* if it satisfies the following two conditions:

1. (*finiteness of proofs*) For every k the set $T(k)=\{\, l \mid \mathrm{Prf}(k,l)\,\}$ is finite. The function from k to the code of $T(k)$ is computable.
2. (*conjoinability of proofs*) For any k and l there is n such that

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

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

Comment 6.3. Every normal proof predicate can be transformed into a single-conclusion one by changing from

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

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

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

Proposition 6.4. *For every normal proof predicate* Prf *there are computable functions* $\mathbf{m}(x, y)$, $\mathbf{a}(x, y)$ *and* $\mathbf{c}(x)$ *such that for all arithmetical formulas* φ, ψ *and all natural numbers* k, n *the following formulas are valid*:

$$\mathrm{Prf}(k, \varphi \to \psi) \wedge \mathrm{Prf}(n, \varphi) \to \mathrm{Prf}(\mathbf{m}(k, n), \psi)$$
$$\mathrm{Prf}(k, \varphi) \to \mathrm{Prf}(\mathbf{a}(k, n), \varphi), \quad \mathrm{Prf}(n, \varphi) \to \mathrm{Prf}(\mathbf{a}(k, n), \varphi)$$
$$\mathrm{Prf}(k, \varphi) \to \mathrm{Prf}(\mathbf{c}(k), \mathrm{Prf}(k, \varphi)).$$

Proof. The following function can be taken as $\mathbf{m}$:

Given k, n *set* $\mathbf{m}(k, n) = \mu z.$“ $\mathrm{Prf}(z, \psi)$ *holds for all* ψ *such that there are* $\ulcorner \varphi \to \psi \urcorner \in T(k)$ *and* $\ulcorner \varphi \urcorner \in T(n)$”.

Likewise, for $\mathbf{a}$ one could take

Given k, n *set* $\mathbf{a}(k, n) = \mu z.$“$T(k) \cup T(n) \subseteq T(z)$”.

Finally, $\mathbf{c}$ may be given by

Given k *set* $\mathbf{c}(k) = \mu z.$“ $\mathrm{Prf}(z, \mathrm{Prf})k, \varphi))$ *for all* $\ulcorner \varphi \urcorner \in T(k)$”.

Such z always exists. Indeed, $\mathrm{Prf}(k, \varphi)$ is a true Δ_1-sentence for every $\ulcorner \varphi \urcorner \in T(k)$, therefore they are all provable in **PA**. Use conjoinability to find a uniform proof of all of them. ⊣

Note that the natural arithmetical proof predicate $\mathrm{Proof}(x, y)$

“x is the code of a derivation containing a formula with the code y”

is an example of a normal proof predicate.

Definition 6.5. An arithmetical *interpretation* $*$ of the **LP**-language has the following parameters:

- a normal proof predicate Prf with the functions $\mathbf{m}(x, y)$, $\mathbf{a}(x, y)$ and $\mathbf{c}(x)$ as in Proposition 6.4,
- an evaluation of propositional letters by sentences of arithmetic,
- an evaluation of proof variables and constants by natural numbers.

Let $*$ commute with boolean connectives,

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

Under an interpretation $*$ a proof polynomial t becomes the natural number t^*, an **LP**-formula F becomes the arithmetical sentence F^*. A formula $(t{:}F)^*$ is always provably Δ_1. Note that **PA** is able to derive any true Δ_1-sentence, and thus to derive a negation of any false Δ_1-sentence (cf. [79]). For a set X of **LP**-formulas by 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 **PA**). An **LP**-formula F is *valid* (with respect to the arithmetical semantics) if F^* is true under all interpretations $*$. F is *provably valid* if $\mathbf{PA} \vdash F^*$ for any interpretation $*$. F is *valid under constant specification $\mathcal{CS}$* if F^* is true under all $\mathcal{CS}$-interpretations $*$. F is *provably valid under constant specification $\mathcal{CS}$* if $\mathbf{PA} \vdash F^*$ for any $\mathcal{CS}$-interpretation $*$. It is obvious that *provably valid* yields *valid*.

PROPOSITION 6.6 (Arithmetical soundness of $\mathbf{LP}_0$).

$$\textit{If } \mathbf{LP}_0 \vdash F \textit{ then } F \textit{ is provably valid.}$$

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

COROLLARY 6.7 (Arithmetical soundness of **LP**).

$$\mathbf{LP}(\mathcal{CS}) \vdash F \Longrightarrow F \textit{ is provably valid under the constant specification } \mathcal{CS}.$$

COMMENT 6.8. The standard provability semantics for **LP** above may be characterized as a *call-by-value* semantics, since the evaluation F^* of a given **LP**-formula F depends upon the value of participating functions. A *call-by-name* provability semantics for **LP** was introduced in [7] and then used in [67], [68], [97], [115]. In the latter semantics F^* depends upon the particular programs for the functions participating in $*$.

§7. A sequent formulation of Logic of Proofs.

By *sequent* we mean a pair $\Gamma \Longrightarrow \Delta$, where Γ and Δ are finite multisets of **LP**-formulas. For Γ, F we mean $\Gamma \cup \{F\}$. To simplify proofs we assume a boolean basis $\to$, $\perp$ and treat the remaining boolean connectives as definable ones.

Axioms of $\mathbf{LPG}_0$ are sequents of the form $\Gamma, F \Longrightarrow F, \Delta$ and $\Gamma, \perp \Longrightarrow \Delta$. Along with the usual Gentzen sequent rules of classical propositional logic, including the cut and contraction rules (e.g., like **G2c** from [104]), the system $\mathbf{LPG}_0$ contains the rules

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

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

$$\frac{\Gamma \Longrightarrow \Delta, s{:}(A \to B) \quad \Gamma \Longrightarrow \Delta, t{:}A}{\Gamma \Longrightarrow \Delta, (s \cdot t){:}B}\,(\Longrightarrow \cdot).$$

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

$$\frac{\Gamma \Longrightarrow A, \Delta}{\Gamma \Longrightarrow c{:}A, \Delta} \quad (\Longrightarrow c),$$

where A is an axiom A0–A4 from §5, and c is a proof constant.

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

PROPOSITION 7.1.
$\mathbf{LPG}_0 \vdash \Gamma \Longrightarrow \Delta$ *if and only if* $\mathbf{LP}_0 \vdash \bigwedge \Gamma \to \bigvee \Delta$,
$\mathbf{LPG} \vdash \Gamma \Longrightarrow \Delta$ *if and only if* $\mathbf{LP} \vdash \bigwedge \Gamma \to \bigvee \Delta$.

The proof proceeds by a straightforward induction both ways.

COROLLARY 7.2. $\mathbf{LP}(\mathcal{CS}) \vdash F$ *if and only if* $\mathbf{LPG}_0 \vdash \mathcal{CS} \Longrightarrow F$.

DEFINITION 7.3. The sequent $\Gamma \Longrightarrow \Delta$ 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 $B \in \Delta$,
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* $t{:}X \in \Delta$.

LEMMA 7.4 (Saturation Lemma). *Suppose* $\mathbf{LPG}_0^- \nvdash \Gamma \Longrightarrow \Delta$. *Then there exists a saturated sequent* $\Gamma' \Longrightarrow \Delta'$ *such that*

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

PROOF. A saturated sequent is obtained by the following *Saturation Algorithm* $\mathcal{A}$. Given $\Gamma \Longrightarrow \Delta$, 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 (undischarged). 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 $X_1, \dots, X_n$ 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 $\mathcal{A}$ always terminates. Indeed, $\mathcal{A}$ is finitely branching and each non-backtracking step breaks either a subformula of $\Gamma \Longrightarrow \Delta$ or a formula of the type $t\!:\!F$, where both t and F occur in $\Gamma \Longrightarrow \Delta$. There are only finitely many of those formulas, which guarantees termination. Moreover, $\mathcal{A}$ terminates with success. Indeed, otherwise $\mathcal{A}$ terminates at the root node $\Gamma \Longrightarrow \Delta$ of the computation tree with all the possibilities exhausted and no way to backtrack. Then the computation tree $\mathcal{T}$ of $\mathcal{A}$ contains the sequent $\Gamma \Longrightarrow \Delta$ at the root, and $\mathbf{LPG}_0$ 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 $\mathbf{LPG}_0^-$, which contradicts the assumption that $\mathbf{LPG}_0^- \nvdash \Gamma \Longrightarrow \Delta$. 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 $\Pi \Longrightarrow \Theta, st\!:\!B$, and its children are 2^n sequents of the form $\Pi \Longrightarrow \Theta, st\!:\!B, Y_1^\sigma, \dots, Y_n^\sigma$, where $\sigma = (\sigma_1 \dots, \sigma_n)$ 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 $X_1, \dots, X_n$ is the list of all formulas such that $X_i \to B$ is a subformula of $\Gamma \Longrightarrow \Delta$. By the induction hypothesis all the child sequents are derivable in $\mathbf{LPG}_0^-$. In particular, among them there are 2^{n-1} pairs of sequents of the form $\Pi \Longrightarrow \Theta', s : (X_1 \to B)$ and $\Pi \Longrightarrow \Theta', t : X_1$. To every such pair apply the rule $(\Longrightarrow \cdot)$ to obtain $\Pi \Longrightarrow \Theta'$ (we assume that $st\!:\!B \in \Theta'$). The resulting 2^{n-1} sequents are of the form $\Pi \Longrightarrow \Theta, st\!:\!B, Y_2^\sigma, \dots, Y_n^\sigma$. After we repeat this procedure $n-1$ more times we end up with the sequent $\Pi \Longrightarrow \Theta$, $st\!:\!B$, which is thus derivable in $\mathbf{LPG}_0^-$. ⊣

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

Lemma 7.5 (Completion Lemma). *For each saturated sequent $\Gamma \Longrightarrow \Delta$ not derivable in $\mathbf{LPG}_0^-$ there is a set of* **LP**-*formulas $\widetilde{\Gamma}$ (a completion of $\Gamma \Longrightarrow \Delta$) such that*

1. *$\widetilde{\Gamma}$ is provably decidable, for each 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. $\Gamma \subseteq \widetilde{\Gamma},\ \Delta \cap \widetilde{\Gamma} = \emptyset$,
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 $t{:}X \in \widetilde{\Gamma}$, 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 **LP**-formulas $\Gamma_0, \Gamma_1, \Gamma_2, \dots$. Let $\Gamma_0 = \{\, F \mid F \in \Gamma \,\}$.

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

- if $i = 3k$, then $\mathcal{COM}$ sets
$$\Gamma_{i+1} = \Gamma_i \cup \bigcup_{s,t} \{\, (s \cdot t){:}Y \mid s{:}(X \to Y), t{:}X \in \Gamma_i \,\},$$
- if $i = 3k+1$, then $\mathcal{COM}$ sets
$$\Gamma_{i+1} = \Gamma_i \cup \bigcup_{t} \{\, !t{:}t{:}X \mid t{:}X \in \Gamma_i \,\},$$
- if $i = 3k+2$, then $\mathcal{COM}$ sets
$$\Gamma_{i+1} = \Gamma_i \cup \bigcup_{s,t} \{\, (s+t){:}X, (t+s){:}X \mid t{:}X \in \Gamma_i, |s| < i \,\}.$$

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 $\mathcal{COM}$; $F \in \Gamma_n$ if and only if $F \in \widetilde{\Gamma}$. A similar argument establishes the finiteness 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 $i = 0, 1, 2, \dots$

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

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}.

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

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 $\Gamma \Longrightarrow \Delta$, 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 $i-1=3k$ 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 $\Gamma_i \subseteq \Gamma_{i+1}$, $Y \in \Gamma_{i+1}$.

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 $X \to Y, X \in \Gamma_{i+1}$. From the description of $\mathcal{COM}$ it follows that $(X \to Y) \in \Gamma$. By the saturation properties of $\Gamma \Longrightarrow \Delta$, 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 7.5 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. ⊣

§8. Completeness theorems.

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

THEOREM 8.1. *The following are equivalent*:

1. $\mathbf{LPG}_0^- \vdash \Gamma \Longrightarrow \Delta$,
2. $\mathbf{LPG}_0 \vdash \Gamma \Longrightarrow \Delta$,
3. $\mathbf{LP}_0 \vdash \bigwedge \Gamma \to \bigvee \Delta$,
4. $\bigwedge \Gamma \to \bigvee \Delta$ *is provably valid in arithmetic*,
5. $\bigwedge \Gamma \to \bigvee \Delta$ *is valid in arithmetic.*

PROOF. The steps from 1 to 2 and from 4 to 5 are trivial. The step from 2 to 3 follows from 7.1, and the step from 3 to 4 follows from 6.6. The only remaining step is thus from 5 to 1. We assume "not 1" and establish "not 5". Suppose $\mathbf{LPG}_0^- \nvdash \Gamma \Longrightarrow \Delta$. Our aim now will be to construct an interpretation $*$ such that $(\bigwedge \Gamma \to \bigvee \Delta)^*$ is false (in the standard model of arithmetic).

From the saturation procedure (Lemma 7.4) get a saturated sequent $\Gamma' \Longrightarrow \Delta'$, and then perform a completion (Lemma 7.5) to get a set of formulas $\widetilde{\Gamma'}$.

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 **LP** and **PA** is injective, i.e.,

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

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

It is clear that if $\mathrm{Prf}(x, y)$ 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 $(s{:}G)^\dagger \equiv \mathrm{Prf}(k, n)$ for the corresponding 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 $\mathrm{Prf}(k_1, n_1)$. 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 **LP**-formula. Moreover, one can construct primitive recursive functions f and g such that

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

Let $(\mathrm{Proof}, \otimes, \oplus, \uparrow)$ be the standard multi-conclusion proof predicate from §6, with $\otimes$ standing for "application", $\oplus$ for "sum" and $\uparrow$ for "proof checker" operations associated with Proof. In particular, for any arithmetical formulas φ, ψ and any natural numbers k, n the following formulas are true:

$$\mathrm{Proof}(k, \varphi \to \psi) \wedge \mathrm{Proof}(n, \varphi) \to \mathrm{Proof}(k \otimes n, \psi),$$
$$\mathrm{Proof}(k, \varphi) \to \mathrm{Proof}(k \oplus n, \varphi),$$
$$\mathrm{Proof}(n, \varphi) \to \mathrm{Proof}(k \oplus n, \varphi),$$
$$\mathrm{Proof}(k, \varphi) \to \mathrm{Proof}(\uparrow k, \mathrm{Proof}(k, \varphi)).$$

Without loss of generality we assume that $\mathrm{Proof}(\ulcorner t \urcorner, k)$ is false for any proof polynomial t and any $k \in \omega$.

Let $\varphi(\vec{y}, z)$ be a provably Σ_1 arithmetical formula. Without loss of generality we assume that $\varphi(\vec{y}, z)$ is provably equivalent to $\exists x\ \psi(x, \vec{y}, z)$ for some provably Δ_1-formula $\psi(x, \vec{y}, z)$. By $\mu z.\varphi(\vec{y}, z)$ we mean a function $z = f(\vec{y})$ that given $\vec{y}$

1. calculates the first pair of natural numbers (k, l) such that $\psi(k, \vec{y}, l)$ holds
2. puts $z = l$.

It is clear that $\mu z.\varphi(\vec{y}, z)$ is computable (though not necessarily total).

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

$$\begin{aligned}\mathrm{Prf}(x, y) \leftrightarrow\ & \mathrm{Proof}(x, y)\\ & \vee (\text{“}x = \ulcorner t \urcorner \textit{ for some } t \textit{ and } y = \ulcorner B^{\dagger} \urcorner \textit{ for some } B \in I(t)\text{”}).\end{aligned}$$

The arithmetical formula “…” above 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 $\mathrm{Proof}(x, y)$ is provably Δ_1. It also follows from FPE that $\mathbf{PA} \vdash \psi$ yields $\mathrm{Prf}(k, \psi)$ for some $k \in \omega$.

We define the arithmetical formulas $M(x, y, z)$, $A(x, y, z)$, $C(x, z)$ as follows. Here s, t denote proof polynomials.

$$\begin{aligned}M(x, y, z) \leftrightarrow\ & (\text{“}x = \ulcorner s \urcorner \textit{ and } y = \ulcorner t \urcorner \textit{ for some } s \textit{ and } t\text{”} \wedge z = \ulcorner s \cdot t \urcorner)\\
& \vee (\text{“}x = \ulcorner s \urcorner \textit{ for some } s \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } t\text{”}\\
& \qquad \wedge \exists v\ [\text{“}v = \mu w.(\textstyle\bigwedge\{\,\mathrm{Proof}(w, B^{\dagger}) \mid B \in I(s)\,\})\text{”}\\
& \qquad\qquad \wedge z = v \otimes y])\\
& \vee (\text{“}x \neq \ulcorner s \urcorner \textit{ for any } s \textit{ and } y = \ulcorner t \urcorner \textit{ for some } t\text{”}\\
& \qquad \wedge \exists v\ [\text{“}v = \mu w.(\textstyle\bigwedge\{\,\mathrm{Proof}(w, B^{\dagger}) \mid B \in I(t)\,\})\text{”}\\
& \qquad\qquad \wedge z = x \otimes v])\\
& \vee (\text{“}x \neq \ulcorner s \urcorner \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } s \textit{ and } t\text{”} \wedge z = x \otimes y).\end{aligned}$$

$$
\begin{aligned}
A(x,y,z) \leftrightarrow{} & (\text{“}x = \ulcorner s \urcorner \textit{ and } y = \ulcorner t \urcorner \textit{ for some } s \textit{ and } t\text{”} \wedge z = \ulcorner s+t \urcorner)\\
& \vee (\text{“}x = \ulcorner s \urcorner \textit{ for some } s \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } t\text{”}\\
& \qquad \wedge \exists v\, [\text{“}v = \mu w.(\textstyle\bigwedge\{\, \mathrm{Proof}(w, B^\dagger) \mid B \in I(s)\,\})\text{”}\\
& \qquad\qquad \wedge z = v \oplus y])\\
& \vee (\text{“}x \neq \ulcorner s \urcorner \textit{ for any } s \textit{ and } y = \ulcorner t \urcorner \textit{ for some } t\text{”}\\
& \qquad \wedge \exists v\, [\text{“}v = \mu w.(\textstyle\bigwedge\{\, \mathrm{Proof}(w, B^\dagger) \mid B \in I(t)\,\})\text{”}\\
& \qquad\qquad \wedge z = x \oplus v])\\
& \vee (\text{“}x \neq \ulcorner s \urcorner \textit{ and } y \neq \ulcorner t \urcorner \textit{ for any } s \textit{ and } t\text{”} \wedge z = x \oplus y).\\
C(x,z) \leftrightarrow{} & (\text{“}x = \ulcorner t \urcorner \textit{ for some } t\text{”} \wedge z = \ulcorner !t \urcorner)\\
& \vee (\text{“}x \neq \ulcorner t \urcorner \textit{ for any } t\text{”}\\
& \qquad \wedge \exists v\, [\text{“}v = \mu w.(\textstyle\bigwedge\{\, \mathrm{Proof}(w, \mathrm{Proof}(t,\varphi) \to \mathrm{Prf}(t,\varphi)) \mid\\
& \qquad\qquad \varphi \in T(t)\,\})\text{”} \wedge z = v \otimes {\uparrow} x]).
\end{aligned}
$$

Here each of "…" denotes a natural arithmetical formula representing in **PA** the corresponding condition. Note that in the definitions of $M(x,y,z)$, $A(x,y,z)$ and $C(x,z)$ above all the functions of sort $\mu w.\varphi$ are provably computable since all the corresponding φ's are provably Δ_1. Therefore $M(x,y,z)$, $A(x,y,z)$ and $C(x,z)$ are provably Σ_1. Let

$$
\begin{aligned}
\mathbf{m}(x,y) &:= \mu z.M(x,y,z),\\
\mathbf{a}(x,y) &:= \mu z.A(x,y,z),\\
\mathbf{c}(x) &:= \mu z.C(x,z).
\end{aligned}
$$

As follows from the above the functions $\mathbf{m}(x,y)$, $\mathbf{a}(x,y)$ and $\mathbf{c}(x)$ are computable. Moreover, Lemma 8.6 below yields that these functions are total.

We continue defining the interpretation $*$. Let Prf for $*$ be the one from *FPE*, and the functions $\mathbf{m}(x,y)$, $\mathbf{a}(x,y)$ and $\mathbf{c}(x)$ are as above.

LEMMA 8.2.

(a) *$t^* = t^\dagger$ for any proof polynomial t,*
(b) *$B^* \equiv B^\dagger$ for any **LP**-formula B.*

PROOF.

(a) Induction on the construction of a proof polynomial. 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 **PA**:

$$
\begin{gathered}
(s\cdot t)^* = \mathbf{m}(s^*, t^*) = \mathbf{m}(\ulcorner s \urcorner, \ulcorner t \urcorner) = \ulcorner s\cdot t \urcorner = (s\cdot t)^\dagger,\\
(s+t)^* = \mathbf{a}(s^*, t^*) = \mathbf{a}(\ulcorner s \urcorner, \ulcorner t \urcorner) = \ulcorner s+t \urcorner = (s+t)^\dagger,\\
(!t)^* = \mathbf{c}(t^*) = \mathbf{c}(\ulcorner t \urcorner) = \ulcorner !t \urcorner = (!t)^\dagger.
\end{gathered}
$$

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

Corollary 8.3. *The mapping $*$ is injective on terms and formulas of* **LP**. *In particular, for all expressions E_1 and E_2*

$$E_1{}^* = E_2{}^* \Longrightarrow E_1 \equiv E_2.$$

Corollary 8.4. *X^* is provably Δ_1 for any* **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 $\mathrm{Prf}(t^*, Y^*)$. By Lemma 8.2,

$$\mathbf{PA} \vdash \mathrm{Prf}(t^*, Y^*) \leftrightarrow \mathrm{Prf}(t, Y^*).$$

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

Lemma 8.5. *If $X \in \widetilde{\Gamma'}$, then* $\mathbf{PA} \vdash X^*$. *If $X \in \Delta'$, then* $\mathbf{PA} \vdash \neg X^*$.

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 if and only if $X \in \widetilde{\Gamma'}$. Let $X = t: Y$ and $t: Y \in \widetilde{\Gamma'}$. Then $\mathbf{PA} \vdash$ "$Y \in I(t)$". By *FPE*, $\mathbf{PA} \vdash \mathrm{Prf}(t, Y^\dagger)$. By Lemma 8.2, $\mathbf{PA} \vdash \mathrm{Prf}(t^*, Y^*)$. Therefore $\mathbf{PA} \vdash (t: Y)^*$.

If $t : Y \in \Delta'$, then $t : Y \notin \widetilde{\Gamma'}$ and "$Y \in I(t)$" is false. The formula $\mathrm{Proof}(t^*, Y^*)$ is also false since t^* is $\ulcorner t \urcorner$ (by Lemma 8.2) and $\mathrm{Proof}(t, k)$ is false for any k by assumption. By *FPE*, $(t{:}Y)^*$ is false. Since $(t{:}Y)^*$ is provably Δ_1 (Lemma 8.4) $\mathbf{PA} \vdash \neg(t{:}Y)^*$.

The induction steps corresponding to boolean connectives are standard and based on the saturation properties of $\Gamma' \Longrightarrow \Delta'$. For example, let $X = Y \to Z \in \widetilde{\Gamma'}$. Then $Y \to Z \in \Gamma'$, and by Definition 7.3, $Y \in \Gamma'$ or $Z \in \Delta'$. By the induction hypothesis, Y^* is true or Z^* is false, and thus $(Y \to Z)^*$ is true, etc. $\dashv$

Lemma 8.6. $\mathbf{PA} \vdash \varphi \iff \mathrm{Prf}(n, \varphi)$ *for some $n \in \omega$.*

Proof. It remains to establish ($\Longleftarrow$). Let $\mathrm{Prf}(n, \varphi)$ hold for some $n \in \omega$. By *FPE*, either $\mathrm{Proof}(n, \varphi)$ holds or $\ulcorner \varphi \urcorner = \ulcorner B^\dagger \urcorner$ for some B such that $t: B \in \widetilde{\Gamma'}$. In the latter case by the saturation property of $\widetilde{\Gamma'}$, $B \in \widetilde{\Gamma'}$. By Lemma 8.5, $\mathbf{PA} \vdash B^*$. By the injectivity of the Gödel numbering, $\varphi \equiv B^\dagger$. By Lemma 8.2, $\varphi \equiv B^*$. Therefore $\mathbf{PA} \vdash \varphi$. $\dashv$

Lemma 8.7. *For all arithmetical formulas φ, ψ and natural numbers k, n the following is true*:

(a) $\mathrm{Prf}(k, \varphi \to \psi) \wedge \mathrm{Prf}(n, \varphi) \to \mathrm{Prf}(\mathbf{m}(k, n), \psi)$.
(b) $\mathrm{Prf}(k, \varphi) \to \mathrm{Prf}(\mathbf{a}(k, n), \varphi)$, $\mathrm{Prf}(n, \varphi) \to \mathrm{Prf}(\mathbf{a}(k, n), \varphi)$.
(c) $\mathrm{Prf}(k, \varphi) \to \mathrm{Prf}(\mathbf{c}(k), \mathrm{Prf}(k, \varphi))$.

PROOF.

(a) Assume $\mathrm{Prf}(k, \varphi \to \psi)$ and $\mathrm{Prf}(n, \varphi)$. There are four possibilities.

(i) Neither of k, n is a Gödel number of a proof polynomial. By *FPE*, both $\mathrm{Proof}(n, \varphi)$ and $\mathrm{Proof}(k, \varphi \to \psi)$ hold, so $\mathrm{Proof}(k \otimes n, \psi)$ does also.

(ii) Both k and n are equal to Gödel numbers of some proof polynomials, say $k = \ulcorner s \urcorner$ and $n = \ulcorner t \urcorner$. By *FPE*, φ is F^* and ψ is G^* for some **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 7.5 (4)), $G \in I(s \cdot t)$. By *FPE*, $\mathrm{Prf}(s \cdot t, G^*)$. By Lemma 8.2 and by definitions, **PA** proves that

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

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

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

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

by the following method. Take $I(t) = \{B_1, \dots, B_l\}$. By definition, $B_i \in \widetilde{\Gamma'}$, $i = 1, \dots, l$. By Lemma 8.5, $\mathbf{PA} \vdash {B_i}^*$ for all $i = 1, \dots, l$. By Lemma 8.2, $\mathbf{PA} \vdash {B_i}^\dagger$ for all $i = 1, \dots, l$. By the conjoinability property of Proof there exists w such that $\mathrm{Proof}(w, {B_i}^\dagger)$ for all $i = 1, \dots, l$. Let j be the least such w. In particular, $\mathrm{Proof}(j, F^\dagger)$. By the definition of $\otimes$, $\mathrm{Proof}(k \otimes j, \psi)$. By the definition of M, $\mathbf{PA} \vdash \mathbf{m}(k, n) = k \otimes j$, therefore $\mathrm{Proof}(\mathbf{m}(k, n), \psi)$ holds.

Case (iv): "s is a Gödel number of a proof polynomial but t is not a Gödel number of any proof polynomial" is similar to (iii).

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

(c) Given $\mathrm{Prf}(k, \varphi)$ there are two possibilities.

(i) $k = \ulcorner t \urcorner$ for some proof polynomial t. By *FPE*, $\varphi \equiv F^\dagger$ for some F such that $F \in I(t)$. By the closure property 7.5 (5) of $\widetilde{\Gamma'}$, $!t : t : F \in \widetilde{\Gamma'}$. By Lemma 8.5, $(!t : t : F)^*$ holds. By definitions,

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

By Lemma 8.2, $t^* = \ulcorner t \urcorner$ and $F^* \equiv F^\dagger$. Therefore $t^* = k$, $F^* \equiv \varphi$ and

$$\mathrm{Prf}(\mathbf{c}(k), \mathrm{Prf}(k, \varphi)).$$

(ii) $k \neq \ulcorner t \urcorner$ for any proof polynomial t. By *FPE*, $\mathrm{Proof}(k, \varphi)$ holds. By definition of the proof checking operation $\uparrow$ for Proof,

$$\mathrm{Proof}(\uparrow k, \mathrm{Proof}(k, \varphi)).$$

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

$$\mu w. \bigwedge\{\, \mathrm{Proof}(w, \mathrm{Proof}(k, \psi) \to \mathrm{Prf}(k, \psi)) \mid \mathrm{Proof}(k, \psi) \,\}.$$

By the definition of l,

$$\mathrm{Proof}(l, \mathrm{Proof}(k, \varphi) \to \mathrm{Prf}(k, \varphi)).$$

Therefore

$$\mathrm{Proof}(l \otimes \uparrow k, \mathrm{Prf}(k, \varphi)).$$

By *FPE*,

$$\mathrm{Prf}(l \otimes \uparrow k, \mathrm{Prf}(k, \varphi)),$$

therefore

$$\mathrm{Prf}(\mathbf{c}(k), \mathrm{Prf}(k, \varphi)). \qquad \dashv$$

LEMMA 8.8. *The normality conditions for* Prf *are fulfilled.*

PROOF. By *FPE*, Prf is provably Δ_1. It follows from *FPE* and 8.6 that for any arithmetical sentence φ

$$\mathbf{PA} \vdash \varphi \quad \text{if and only if} \quad \mathrm{Prf}(n, \varphi) \text{ holds for some } n.$$

Finiteness of proofs. For each k the set

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

is finite. Indeed, if k is a Gödel number of a proof polynomial, we can use the finiteness of $I(t)$; otherwise we use the normality of Proof. An algorithm for the function from k to the code of $T(k)$ for Prf can be easily constructed from those for Proof, and from the decision algorithm for $I(t)$, Lemma 7.5 (1).

Conjoinability of proofs for Prf is realized by the function $\mathbf{a}(x, y)$, since by Lemma 8.7,

$$T(k) \cup T(n) \subseteq T(\mathbf{a}(k, n)). \qquad \dashv$$

Let us finish the proof of the final "not 1 implies not 5" part of 8.1. Given a sequent $\Gamma \Longrightarrow \Delta$ not provable in $\mathbf{LPG}_0^-$ we have constructed an interpretation $*$ such that Γ^* are all true, and Δ^* are all false in the standard model of arithmetic (8.5). Therefore, $(\bigwedge \Gamma \to \bigvee \Delta)^*$ is false. $\dashv$

COROLLARY 8.9 (Completeness of **LP**).

$$\mathbf{LP}(\mathcal{CS}) \vdash F \iff F \textit{ is (provably) valid under constant specification } \mathcal{CS}.$$

COROLLARY 8.10.

$$\mathbf{LP} \vdash F \iff F \textit{ is (provably) valid under some constant specification}.$$

COROLLARY 8.11. $\mathbf{LP}_0$ *is decidable.*

Given an **LP**-formula F run the saturation algorithm $\mathcal{A}$ on a sequent $\Longrightarrow F$. If $\mathcal{A}$ fails, then $\mathbf{LP}_0 \vdash F$. Otherwise, $\mathbf{LP}_0 \nvdash F$.

COROLLARY 8.12 (Cut elimination in $\mathbf{LP}_0$). *Every sequent that is derivable in* $\mathbf{LPG}_0$ *can be derived without the cut rule.*

Proof. By Theorem 8.1 $\mathbf{LPG}_0^- \vdash \Gamma \Longrightarrow \Delta$ if and only if $\mathbf{LPG}_0 \vdash \Gamma \Longrightarrow \Delta$. ⊣

Corollary 8.13 (Cut elimination in **LP**). *Every sequent derivable in* **LPG** *can be derived without the cut rule.*

Proof. Cut elimination for **LP** can be established by a direct system of reductions, and it has been done in [9], [12]. We may also get the cut elimination theorem for **LP** as a side product of the arithmetical completeness theorem for **LP**. Indeed, a straightforward analogue of Theorem 8.1 where $\mathbf{LP}_0$ and $\mathbf{LPG}_0$ are replaced by **LP** and **LPG** respectively holds. As in 8.1 it suffices to establish that if $\mathbf{LPG} \nvdash \Gamma \Longrightarrow \Delta$ then for any constant specification $\mathcal{CS}$ there exists a $\mathcal{CS}$-interpretation $*$ such that the arithmetical sentence $(\bigwedge \Gamma \rightarrow \bigvee \Delta)^*$ is false. Let us sketch changes that should be made in the definitions and proofs from §7 and §8 to make them work for **LP**. Fix a constant specification $\mathcal{CS}$. Definition 7.3 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 7.4 holds with the new definition of a saturated sequent and $\mathbf{LPG}^-$ instead of $\mathbf{LPG}_0^-$. Item 3 of Lemma 7.5 should be read as

3. $\mathcal{CS} \in \widetilde{\Gamma}$ and if $t{:}X \in \widetilde{\Gamma} - \mathcal{CS}$, 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 7.5 and the entire 8.1 remain intact under the new definitions. ⊣

Comment 8.14. Decidability of **LP** follows from the results of [82]. This fact can also be easily obtained from the cut elimination property of **LP** (Corollary 8.13).

Corollary 8.15 (Non-emptiness of provability semantics for **LP**). *For any constant specification* $\mathcal{CS}$ *there exists a* $\mathcal{CS}$*-interpretation* $*$.

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

From $\mathbf{LPG}_0 \nvdash \mathcal{CS} \Longrightarrow$ it follows that $\mathbf{LPG}_0 \nvdash\ \Longrightarrow \neg\mathcal{CS}$. By 8.1, there exists an interpretation $*$ such that $(\neg\mathcal{CS})^*$ is false, i.e., $\mathcal{CS}^*$ is true. ⊣

§9. Realization of modal and intuitionistic logics. It is easy to see that the forgetful projection of **LP** is **S4**-compliant. Let F^o be the result of substituting $\Box X$ for all occurrences of $t{:}X$ in F.

LEMMA 9.1. *If* **LP** $\vdash F$, *then* **S4** $\vdash F^o$.

PROOF. This is a straightforward induction on a derivation in **LP**. ⊣

The goal of the current section is to establish the converse, namely that **LP** suffices to realize any theorem of **S4**.

DEFINITION 9.2. By an **LP**-*realization* of a modal formula F we mean an assignment of proof polynomials to all occurrences of the modality in F along with a constant specification of all constants occurring in those proof polynomials. By F^r we understand 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. the 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 $\Gamma \Longrightarrow \Delta, F$ 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 $F, \Gamma \Longrightarrow \Delta$ 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, whereas 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.

DEFINITION 9.3. A realization r is *normal* if all negative occurrences of $\Box$ are realized by proof variables and the corresponding constant specification is injective.

THEOREM 9.4. *If* **S4** $\vdash F$ *then* **LP** $\vdash F^r$ *for some normal realization* r.

PROOF. Consider a cut-free sequent formulation of **S4** (cf. [19], [80]), with sequents $\Gamma \Longrightarrow \Delta$, where Γ and Δ are finite multisets of modal formulas. Axioms are sequents of the form $S \Longrightarrow S$, where S is a propositional letter, and the sequent $\perp \Longrightarrow$. Along with the usual structural rules (weakening, contraction, cut) and rules introducing boolean connectives there are two proper modal rules:

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

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

If $\mathbf{S4} \vdash F$, then there exists a cut-free derivation $\mathcal{T}$ of a sequent $\Longrightarrow F$. It suffices now to construct a normal realization r with an injective constant specification $\mathcal{CS}$ such that $\mathbf{LP}(\mathcal{CS}) \vdash \bigwedge \Gamma^r \rightarrow \bigvee \Delta^r$ for any sequent $\Gamma \Longrightarrow \Delta$ in $\mathcal{T}$. We will also speak about a sequent $\Gamma \Longrightarrow \Delta$ being derivable in $\mathbf{LP}$ meaning $\mathbf{LP} \vdash \bigwedge \Gamma \rightarrow \bigvee \Delta$, or, equivalently, $\mathbf{LPG} \vdash \Gamma \Longrightarrow \Delta$. Note that in $\mathcal{T}$ the rules respect polarities, all occurrences of $\Box$ introduced by $(\Longrightarrow \Box)$ are positive, and all negative occurrences are introduced by $(\Box \Longrightarrow)$ 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 instance of the $(\Longrightarrow \Box)$ 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 $\Box B$ by "$x : B$" for a fresh proof variable x.

Step 2. Pick an essential family f, enumerate all the occurrences of rules $(\Longrightarrow \Box)$ 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 polynomial

$$v_1 + \cdots + v_{n_f},$$

where v_i's are fresh provisional variables. The resulting tree $\mathcal{T}'$ is labelled by $\mathbf{LP}$-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. At the initial moment $\mathcal{CS}$ is empty. By induction on the depth of a node in $\mathcal{T}'$ we establish that after the process passes a node, the sequent assigned to this node becomes derivable in $\mathbf{LP}(\mathcal{CS})$. The axioms $S \Longrightarrow S$ and $\bot \Longrightarrow$ are derivable in $\mathbf{LP}_0$. For every rule other than $(\Longrightarrow \Box)$ we change neither the realization of formulas nor $\mathcal{CS}$, and just establish that the concluding sequent is provable in $\mathbf{LP}(\mathcal{CS})$ given that the premises are. Moreover, every move down in the tree $\mathcal{T}'$ other than $(\Longrightarrow \Box)$ is a rule of the system $\mathbf{LPG}$. Therefore, the induction steps corresponding to these moves follow easily from the equivalence of $\mathbf{LP}$ and $\mathbf{LPG}$.

Let an occurrence of the rule $(\Longrightarrow \Box)$ have number i in the numbering of all rules $(\Longrightarrow \Box)$ from a given family f. The corresponding node in $\mathcal{T}'$ is labelled by

$$\frac{y_1 : B_1, \ldots, y_k : B_k \Longrightarrow B}{y_1 : B_1, \ldots, y_k : B_k \Longrightarrow (u_1 + \ldots + u_{n_f}) : B}\ ,$$

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 $y_1 : B_1, \ldots, y_k : B_k \Longrightarrow B$ is derivable in $\mathbf{LP}(\mathcal{CS})$. By the Lifting Lemma 5.4, construct a proof polynomial $t(y_1, \ldots, y_n)$ and extend the constant specification to get a new injective $\mathcal{CS}$ such that

$$\mathbf{LP}(\mathcal{CS}) \vdash y_1 : B_1, \ldots, y_k : B_k \Longrightarrow t(y_1, \ldots, y_n) : B.$$

Since

$$\mathbf{LP}_0 \vdash t : B \rightarrow (u_1 + \cdots + u_{i-1} + t + u_{i+1} + \cdots + u_{n_f}) : B$$

we have

$$\mathbf{LP}(\mathcal{CS}) \vdash y_1 : B_1, \ldots, y_k : B_k \\ \Longrightarrow (u_1 + \cdots + u_{i-1} + t + u_{i+1} + \cdots + u_{n_f}) : B.$$

Now substitute $t(y_1, \ldots, y_n)$ for u_i everywhere in $\mathcal{T}'$ and $\mathcal{CS}$. The latter remains injective after such a substitution though this may lead to constant specifications of the sort $c : A(c)$ where $A(c)$ contains c.

Note that $t(y_1, \ldots, y_n)$ has no provisional variables, and that there is one less provisional variable (namely u_i) in $\mathcal{T}'$ and $\mathcal{CS}$. The conclusion of the given rule ($\Longrightarrow \Box$) becomes derivable in $\mathbf{LP}(\mathcal{CS})$, and the induction step is complete.

Eventually, we substitute polynomials of non-provisional variables for all provisional variables in $\mathcal{T}'$ and $\mathcal{CS}$ and establish that the root sequent of $\mathcal{T}'$ is derivable in $\mathbf{LP}(\mathcal{CS})$. The realization r built by this procedure is normal. $\dashv$

Corollary 9.5 (Realization of **S4**).

$$\mathbf{S4} \vdash F \iff \mathbf{LP} \vdash F^r \textit{ for some realization } r.$$

Corollary 9.6 (Arithmetical completeness of **S4**).

$$\mathbf{S4} \vdash F \iff F^r \textit{ is (provably) valid for some realization } r.$$

Comment 9.7. **S4**-theorems admit essentially different realizations in **LP**. For example, among possible realizations of $\Box F \vee \Box F \rightarrow \Box F$ there are

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

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

Modal formulas can be realized by some restricted classes of proof polynomials. For example, the standard realization of the **S4**-theorem $(\Box A \vee \Box B) \rightarrow \Box(A \vee B)$ gives $(x : A \vee y : B) \rightarrow (a \cdot x + b \cdot y) : (A \vee B)$ with the injective constant specification $a : (A \rightarrow A \vee B)$, $b : (B \rightarrow A \vee B)$. The same modal formula can be realized in **LP** as $(c : A \vee c : B) \rightarrow (c \cdot c) : (A \vee B)$ with the constant specification $c : (A \rightarrow A \vee B)$, $c : (B \rightarrow A \vee B)$. However, one needs the whole expressive power of **LP** to maintain injectivity of constant specifications and to distinguish between apparently unrelated proofs (like $x{:}A$ and $y{:}B$ above).

DEFINITION 9.8. A propositional formula F is *proof realizable* if $(t(F))^r$ is valid under some realization r.

THEOREM 9.9 (Provability completeness of **Int**). *For any formula F*

$$\mathbf{Int} \vdash F \iff F \textit{ is proof realizable.}$$

PROOF. A straightforward combination of

$$\mathbf{Int} \vdash F \iff \mathbf{S4} \vdash t(F)$$

([33] section 3.9, [44], [77], [104] sections 10.2 and 10.6), and

$$\mathbf{S4} \vdash t(F) \iff (t(F))^r \textit{ is valid for some realization } r$$

(Corollary 9.6). ⊣

COMMENT 9.10. Theorem 9.9 provides an exact specification of **Int** by means of classical notion of proof consistent with *BHK* semantics.

In addition to Gödel's translation $t(F)$ one could consider *McKinsey-Tarski translation* that prefixes only atoms and implications in F. A result similar to Theorem 9.9 holds for proof realizability based on the McKinsey-Tarski translation too.

§10. Realization of λ-calculi.

Through a realization in **LP** both modality and λ-terms receive a uniform provability semantics.

Assume that a calculus of λ-terms is presented as the sequent calculus of the format

$$x_1{:}A_1, \dots, x_n{:}A_n \Longrightarrow t(\vec{x}){:}B$$

with the reading

$$\textit{term } t(\vec{x}) \textit{ has type } B \textit{ provided } x_i \textit{ has type } A_i \textit{ for } i = 1, 2, \dots, n$$

(cf. system **G2i*** from [104]). Under such formulation a λ-term is represented by a sequent, formation rules of λ-terms become inference rules in the corresponding sequent calculus.

A straightforward observation shows that some of the λ-term constructors can be naturally represented as derivations in **LPG**. For example, the pairing function introduction rule

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

has a natural counterpart **LPG**-derivation

$$\dfrac{\dfrac{\overline{\Gamma \Longrightarrow c:(A \to (B \to (A \wedge B))}\qquad \Gamma \Longrightarrow t:A}{\Gamma \Longrightarrow (c\cdot t):(B \to (A \wedge B))}\qquad \Gamma \Longrightarrow s:B}{\Gamma \Longrightarrow (c\cdot t\cdot s):(A \wedge B)}\,.$$

In fact the entire λ-calculus can be embedded into **LPG** ([9], [12]). The key element of this embedding is emulating λ-abstraction which can be done in many different ways, for example, in the Curry style ([9], [12], cf. [104], pp. 17–18). In either way we find a kind of admissible rule in **LPG**, which represents the λ-abstraction.

$$\frac{\vec{y}:\Gamma, x:A \Longrightarrow t(x):B}{\vec{y}:\Gamma \Longrightarrow \lambda^* x.t(x):(A \to B)}\ \ (x\ \textit{does not occur in}\ \vec{y}:\Gamma, A, B),$$

where $\lambda^* x.t(x)$ is a proof polynomial not containing x built from the given derivation of $\vec{y}:\Gamma, x:A \Longrightarrow t(x):B$.

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

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

is naturally represented by the following transformation of derivations in **LPG**:

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

transforms into

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

The rule of *α-conversion* corresponds to an obviously valid rule of renaming bounded variables in **LPG**-derivations with abstraction.

Since modal logic **S4** and all standard λ-term constructors can be represented by proof polynomials, the Logic of Proofs can also emulate modal λ-calculi. As it was shown in [9], [12] the intuitionistic version of **LP** naturally realizes the modal λ-calculus for **IS4** ([25], [75], [91], cf. also [30]) and thus supplies modal λ-terms with the standard provability semantics. This may be considered as a more general abstract version of the Curry-Howard isomorphism which relates terms/types with proofs/formulas.

§11. First order case. Theories based on the first order modal logics were studied in [14], [40], [41], [49], [50], [55], [81], [85], [86], [93], [95], [96], and many other papers.

In the first order logic of proofs constants and proof letters depend on individual variables: $u(\vec{x}), c(\vec{x}), \ldots$ and are interpreted as provably recursive arithmetical terms. Here are some examples of valid principles accompanied by their plain modal projections.

$c(y){:}(\forall x\, A(x) \rightarrow A(y))$	$\Box(\forall x\, A(x) \rightarrow A(y))$
$u{:}\forall x\, A(x) \rightarrow (c(y)\cdot u){:}A(y)$	$\Box\forall x\, A(x) \rightarrow \Box A(y)$
$u{:}\forall x\, A(x) \rightarrow \forall y\, ((c(y)\cdot u){:}A(y))$	$\Box\forall x\, A(x) \rightarrow \forall y\, \Box A(y)$

In [15] it was shown that the first order logic of proofs is hyperarithmetical (in fact, $\Pi^0_1(\mathbf{TA})$-complete, where **TA** is the set of all true arithmetical sentences). In particular, this means that this logic does not admit a complete axiomatization. The logic of proofs based on the standard Gödel numbering were studied in [6], [17], [116].

§12. Discussion.

1. There are two provability models each having its own areas of applications:

A. *The logic* **L** *of formal provability* ([27], [29], [98], [100]) *with the "non-reflexive" Löb principle* $\Box(\Box F \rightarrow F) \rightarrow \Box F$. Within this model proofs are represented implicitly by existential quantifiers. The highlights of this model are formalizations of the second Gödel incompleteness theorem, Löb theorem and fixed point theorem in the propositional language. **L** finds decent applications in traditional proof theory (cf. [3], [23], [37], [113]).

B. *Gödel's provability calculus* **S4** *and its decoding* **LP** *with the reflection principle* $\Box F \rightarrow F$. Within this model proofs are represented explicitly by computable terms. This model gives solutions to the Gödel provability calculus problem and to the classical *BHK* problem. The **S4**/**LP** explicit provability model has reached out to such areas as constructive semantics, modal logics and logics of knowledge, combinatory logic and λ-calculus, automated deduction and formal verification, etc.

The models A and B are compatible because they are both based on the Gödelian provability. The joint logic of proofs and formal provability has been found in [97], [115].

2. A recent application of explicit provability model: stability of verification. In the framework of formal provability the stability of verification systems is not internally provable ([1], [36]). Rather the reflexive provability model provides a verification mechanism with provable stability ([10]) thus fixing a certain loophole in the foundations of verification.

3. The format *t is a proof of F* introduced by Gödel in [46] yields proof polynomials (cf. [7], [11]) and the whole of **LP** and **S4**. Therefore, there

is nothing arbitrary in those systems. The corresponding completeness theorems demonstrate that nothing is missing in **LP** and **S4** either.

4. Proof polynomials reveal the explicit content of modality and provide a fresh look at modal logic and its applications in general. **S4** is a lazy higher level language on top of **LP**. Explicit counterparts of modal logics **K**, **K4**, and **S5** were found in [13], [31].

5. **LP** may be regarded as a basic epistemic logic with explicit justifications; a problem of finding such systems was raised by van Benthem in [109].

6. The complexity of the Logic of Proofs has been studied in [69] where it was shown that the known upper bounds on the decision procedure for **LP** are much better (Σ_2^{P} in the polynomial hierarchy) than the ones for **S4** or **Int** (PSPACE).

7. Gabbay's Labelled Deductive Systems ([42]) may serve as a natural framework for **LP**. Intuitionistic Type Theory ([73], [74]) and the second order λ-calculus ([43]) use the format $t : F$ with its informal provability reading and could benefit from the corresponding formal semantics.

Acknowledgements. This work has benefited from many interactions over the past several years with S. Abramsky, S. Adian, J. Alt, A. Avron, M. Baaz, H. Barendregt, L. Beklemishev, J. van Benthem, C. Bernardi, G. Boolos, S. Buss, A. Chagrov, R. Constable, D. van Dalen, N. Dershowitz, J. Diller, J.M. Dunn, E. Engeler, L. Esakia, S. Feferman, H. Friedman, D. Gabbay, R. Gandy, J.-Y. Girard, G. Gottlob, Y. Gurevich, P. Hajek, L. Harrington, J. Hartmanis, W. Hodges, G. Jäger, G. Japaridze, D. de Jongh, M. Kanovich, P. Kolaitis, A. Kopylov, D. Kozen, V. Krupski, A. Leisch, V. Lifshits, A. Macintyre, S. MacLane, L. Maksimova, V. McGee, G. Mints, F. Montagna, Y. Moschovakis, P. Naumov, A. Nerode, A. Nogin, E. Nogina, P. Odifreddi, R. Parikh, C. Parsons, W. Pohlers, V. Pratt, A. Preller, J. Remmel, D. Roorda, G. Sacks, G. Sambin, A. Scedrov, K. Segerberg, V. Shavkurov, V. Shekhtman, H. Schwichtenberg, N. Shankar, R. Shore, T. Sidon-Yavorskaya, T. Slaman, R. Soare, T. Strassen, W. Tait, B. Trakhtenbrot, A. Troelstra, V. Uspensky, M. Vardi, A. Voronkov, A. Visser, S. Weinstein, R. Yavorsky, and M. Zakharyaschev.

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-Yavorskaya, and Fred Smith for reading of different versions of this paper which led to valuable improvements.

REFERENCES

[1] S. Allen, R. Constable, D. Howe, and W. Aitken, *The semantics of reflected proofs*, ***Proceedings of the fifth annual IEEE Symposium on Logic in Computer Science*** (Los Alamitos, California, USA), IEEE Computer Society Press, 1990, pp. 95–107.

[2] J. ALT and S. ARTEMOV, *Reflective λ-calculus*, ***Technical Report CFIS 2000-06***, Cornell University, 2000.

[3] S. ARTEMOV, *Applications of modal logic in proof theory*, ***Voprosy Kibernetiki: Nonclassical logics and application***, Nauka, Moscow, 1982, (in Russian), pp. 3–20.

[4] ———, *Non-arithmeticity of truth predicate logics of provability*, ***Soviet Mathematics Doklady***, vol. 32 (1985), no. 2, pp. 403–405.

[5] ———, *Kolmogorov logic of problems and a provability interpretation of intuitionistic logic*, ***Theoretical aspects of reasoning about knowledge—III Proceedings***, Morgan Kaufman Pbl., 1990, pp. 257–272.

[6] ———, *Logic of proofs*, ***Annals of Pure and Applied Logic***, vol. 67 (1994), no. 1, pp. 29–59.

[7] ———, *Operational modal logic*, ***Technical Report MSI 95-29***, Cornell University, 1995.

[8] ———, *Proof realizations of typed λ-calculi*, ***Technical Report MSI 95-02***, Cornell University, 1997.

[9] ———, *Logic of proofs: a unified semantics for modality and λ-terms*, ***Technical Report CFIS 98-06***, Cornell University, 1998.

[10] ———, *On explicit reflection in theorem proving and formal verification*, ***Automated deduction—CADE-16. Proceedings of the 16th International Conference on Automated Deduction, Trento, Italy, July 1999***, Lecture Notes in Artificial Intelligence, vol. 1632, Springer-Verlag, 1999, pp. 267–281.

[11] ———, *Operations on proofs that can be specified by means of modal logic*, ***Advances in modal logic***, vol. 2, CSLI Publications, Stanford University, 2001.

[12] ———, *Unified semantics for modality and λ-terms via proof polynomials*, ***Logic, Language and Computation. Volume 3***, CSLI Publications, Stanford University, (to appear).

[13] S. ARTEMOV, E. KAZAKOV, and D. SHAPIRO, *Epistemic logic with justifications*, ***Technical Report CFIS 99-12***, Cornell University, 1999.

[14] S. ARTEMOV and F. MONTAGNA, *On first order theories with provability operator*, ***The Journal of Symbolic Logic***, vol. 59 (1994), no. 4, pp. 1139–1153.

[15] S. ARTEMOV and T. SIDON-YAVORSKAYA, *On the first order logic of proofs*, ***Technical Report CFIS 99-11***, Cornell University, 1999.

[16] S. ARTEMOV and T. STRASSEN, *Functionality in the basic logic of proofs*, ***Technical Report IAM 92-004***, Department of Computer Science, University of Bern, Switzerland, 1992.

[17] ———, *The logic of the Gödel proof predicate*, ***Computational logic and proof theory. Proceedings of the Third Kurt Gödel Colloquium, Brno, August 1993*** (G. Gottlob, A.Leitsch, and D. Mundici, editors), Lecture Notes in Computer Science, vol. 713, Springer-Verlag, 1993, pp. 71–82.

[18] J. AVIGAD and S. FEFERMAN, *Gödel's functional ("Dialectica") interpretation*, ***Handbook of proof theory*** (S. Buss, editor), Elsevier, 1998, pp. 337–406.

[19] A. AVRON, *On modal systems having arithmetical interpretation*, ***The Journal of Symbolic Logic***, vol. 49 (1984), pp. 935–942.

[20] H. BARENDREGT, *Lambda calculi with types*, ***Handbook of logic in computer science*** (S. Abramsky, D.M. Gabbay, and T.S.E. Maibaum, editors), vol. 2, Oxford University Press, 1992, pp. 118–309.

[21] G. BARTHE, J. HATCLIFF, and M. SØRENSEN, *A notion of a classical pure type system*, ***Electronic Notes in Theoretical Computer Science***, vol. 6 (1997), Proceedings of MFPS'97. (S. Brookes and M. Mislove, editors).

[22] M. BEESON, ***Foundations of constructive mathematics***, Springer-Verlag, 1980.

[23] L. BEKLEMISHEV, *Parameter-free induction and provably total computable functions*, ***Theoretical Computer Science***, vol. 224 (1999), no. 1-2, pp. 13–33.

[24] E.W. Beth, *Semantic construction of intuitionistic logic*, ***Kon. Nederl. Akad. Wetensch. Afd. Let. Med., Nieuwe Serie***, vol. 19/11 (1956), pp. 357–388.

[25] G. Bierman and V. de Paiva, *Intuitionistic necessity revisited*, ***Proceedings of the Logic at Work conference, Amsterdam, 1992***, 1996, Second revision, `http://theory.doc.ic.ac.uk/tfm/papers.html`.

[26] G. Birkhoff, *On the structure of abstract algebras*, ***Proceedings of the Cambridge Philosophical Society***, vol. 31 (1935), pp. 433–454.

[27] G. Boolos, ***The unprovability of consistency: An essay in modal logic***, Cambridge University Press, 1979.

[28] ———, *The logic of provability*, ***American Mathematical Monthly***, vol. 91 (1984), pp. 470–480.

[29] ———, ***The logic of provability***, Cambridge University Press, 1993.

[30] 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.

[31] V. Brezhnev, *On explicit counterparts of modal logics*, ***Technical Report CFIS 2000-05***, Cornell University, 2000.

[32] S. Buss, *The modal logic of pure provability*, ***Notre Dame Journal of Formal Logic***, vol. 31 (1990), no. 2, pp. 225–231.

[33] A. Chagrov and M. Zakharyaschev, ***Modal logic***, Oxford Science Publications, 1997.

[34] R. Constable, *Types in logic, mathematics and programming*, ***Handbook of proof theory*** (S. Buss, editor), Elsevier, 1998, pp. 683–786.

[35] H. B. Curry and R. Feys, ***Combinatory logic***, North-Holland, Amsterdam, 1958.

[36] M. Davis and J. Schwartz, *Metamathematical extensibility for theorem verifiers and proof checkers*, ***Computers and Mathematics with Applications***, vol. 5 (1979), pp. 217–230.

[37] D. de Jongh and G. Japaridze, *Logic of provability*, ***Handbook of proof theory*** (S. Buss, editor), Elsevier, 1998, pp. 475–546.

[38] S. Feferman, *A language and axioms for explicit mathematics*, ***Algebra and logic*** (J. N. Crossley, editor), Springer-Verlag, 1975, pp. 87–139.

[39] ———, *Constructive theories of functions and classes*, ***Logic colloquium '78*** (M. Boffa, D. van Dalen, and K. McAloon, editors), North-Holland, 1979, pp. 159–224.

[40] R. Flagg, *Church's Thesis is consistent with epistemic arithmetic*, ***Intensional mathematics*** (S. Shapiro, editor), North-Holland, 1985, pp. 121–172.

[41] R. Flagg and H. Friedman, *Epistemic and intuitionistic formal systems*, ***Annals of Pure and Applied Logic***, vol. 32 (1986), no. 1, pp. 53–60.

[42] D. M. Gabbay, ***Labelled deductive systems***, Oxford University Press, 1994.

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

[44] K. Gödel, *Eine Interpretation des intuitionistischen Aussagenkalküls*, ***Ergebnisse Math. Colloq.***, vol. 4 (1933), pp. 39–40.

[45] ———, *Über eine bisher noch nicht benütztwe Erweiterung des finiten Standpunktes*, ***Dialectica***, vol. 12 (1958), pp. 280–287.

[46] ———, *Vortrag bei Zilsel, 1938*, ***Kurt Gödel Collected Works*** (S. Feferman, editor), vol. III, Oxford University Press, 1995, pp. 86–113.

[47] R. Goldblatt, *Arithmetical necessity, provability and intuitionistic logic*, ***Theoria***, vol. 44 (1978), pp. 38–46.

[48] ———, ***Topoi***, North-Holland, 1979.

[49] N. D. Goodman, *Epistemic arithmetic is a conservative extension of intuitionistic arithmetic*, ***The Journal of Symbolic Logic***, vol. 49 (1984), pp. 192–203.

[50] ———, *A genuinely intensional set theory*, ***Intensional mathematics*** (S. Shapiro, editor), North-Holland, 1985, pp. 63–79.

[51] N.D. Goodman, *A theory of constructions is equivalent to arithmetic*, ***Intuitionism and proof theory*** (J. Myhill, A. Kino, and R. E. Vesley, editors), North-Holland, 1970, pp. 101–120.

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

[53] ———, *Die intuitionistische Grundlegung der Mathematik*, ***Erkenntnis***, vol. 2 (1931), pp. 106–115.

[54] ———, ***Mathematische Grundlagenforschung. Intuitionismus. Beweistheorie***, Springer-Verlag, Berlin, 1934.

[55] J. Hintikka, ***Knowledge and Belief***, Cornell University Press, Ithaca, 1962.

[56] S. Kleene, *On the interpretation of intuitionistic number theory*, ***The Journal of Symbolic Logic***, vol. 10 (1945), no. 4, pp. 109–124.

[57] ———, *Classical extensions of intuitionistic mathematics*, ***Logic, methodology and philosophy of science 2*** (Y. Bar-Hillel, editor), North-Holland, 1965, pp. 31–44.

[58] A. Kolmogoroff, *Zur Deutung der intuitionistischen Logik*, ***Mathematische Zeitschrift***, vol. 35 (1932), pp. 58–65, English translation in ***Selected works of A.N. Kolmogorov. Volume I: Mathematics and Mechanics***, (V.M. Tikhomirov, editor).

[59] A. Kolmogorov, *About my papers on intuitionistic logic*, ***Selected works of A. N. Kolmogorov. Volume I: Mathematics and mechanics*** (V. M. Tikhomirov, editor), Kluwer, 1985, pp. 451–452.

[60] D. Kozen and J. Tiuryn, *Logic of programs*, ***Handbook of theoretical computer science. Volume B, Formal models and semantics*** (J. van Leeuwen, editor), Elsevier, 1990, pp. 789–840.

[61] M. Kracht, ***Tools and techniques in modal logic***, Elsevier, 1999.

[62] G. Kreisel, *Foundations of intuitionistic logic*, ***Logic, methodology and philosophy of science. Proceedings of the 1960 International Congress*** (E. Nagel, P. Suppes, and A. Tarski, editors), Stanford University Press, 1962, pp. 198–210.

[63] ———, *On weak completeness of intuitionistic predicate logic*, ***The Journal of Symbolic Logic***, vol. 27 (1962), pp. 139–158.

[64] ———, *Mathematical logic*, ***Lectures in Modern Mathematics III*** (T. L. Saaty, editor), Wiley and Sons, New York, 1965, pp. 95–195.

[65] S. Kripke, *Semantical considerations on modal logic*, ***Acta Philosophica Fennica***, vol. 16 (1963), pp. 83–94.

[66] ———, *Semantical analysis of intuitionistic logic. I*, ***Formal systems and recursive functions. Proceedings of the 8th Logic Colloquium*** (J. N. Crossley and M. A. E. Dummett, editors), North-Holland, 1965, pp. 92–130.

[67] V. Krupski, *Operational logic of proofs with functionality condition on proof predicate*, ***Logical foundations of Computer Science '97, Yaroslavl'*** (S. Adian and A. Nerode, editors), Lecture Notes in Computer Science, vol. 1234, Springer-Verlag, 1997, pp. 167–177.

[68] ———, *The single-conclusion proof logic and inference rules specification*, ***Annals of Pure and Applied Logic***, (to appear in 2001), in the volume on the conference ***St. Petersburg Days of Logic and Computability, 1999***, (Yu. Matiyasevich, editor).

[69] A. Kuznets, *On the complexity of explicit modal logics*, ***Computer Science Logic 2000***, Lecture Notes in Computer Science, vol. 1862, Springer-Verlag, 2000, pp. 371–383.

[70] A. Kuznetsov and A. Muravitsky, *The logic of provability*, ***Abstracts of the 4th All-union conference on mathematical logic***, 1976, in Russian, p. 73.

[71] H. Läuchli, *An abstract notion of realizability for which intuitionistic predicate logic is complete*, ***Intuitionism and proof theory*** (J. Myhill, A. Kino, and R. E. Vesley, editors), North-Holland, 1970, pp. 227–234.

[72] E. LEMMON, *New foundations for Lewis's modal systems*, ***The Journal of Symbolic Logic***, vol. 22 (1957), pp. 176–186.

[73] P. MARTIN-LÖF, *Constructive mathematics and computer programming*, ***Logic, methodology and philosophy of science VI*** (L. J. Cohen, J. Łøs, H. Pffeifer, and K.-P. Podewski, editors), North-Holland, 1982, pp. 153–175.

[74] ———, ***Intuitionistic type theory***, Bibliopolis, Naples, 1984.

[75] S. MARTINI and A. MASINI, *A computational interpretation of modal proofs*, ***Proof theory of modal logics. workshop proceedings*** (H. Wansing, editor), Kluwer, 1994.

[76] J. C. C. MCKINSEY and A. TARSKI, *On closed elements of closure algebras*, ***Annals of Mathematics***, vol. 47 (1946), pp. 122–162.

[77] ———, *Some theorems about the sentential calculi of Lewis and Heyting*, ***The Journal of Symbolic Logic***, vol. 13 (1948), pp. 1–15.

[78] YU. MEDVEDEV, *Finite problems*, ***Soviet Mathematics Doklady***, vol. 3 (1962), pp. 227–230.

[79] E. MENDELSON, ***Introduction to mathematical logic***, Wadsworth, 1987.

[80] G. MINTS, *Lewis' systems and system T (a survey 1965–1973).*, ***Feys. Modal logic (Russian translation)***, Nauka, Moscow, 1974, (in Russian). English translation in G. MINTS, ***Selected papers in proof theory***, Bibliopolis, Napoli, 1992, pp. 422–509.

[81] ———, *On Novikov's hypothesis*, ***Modal and intensional logics***, Nauka, Moscow, 1978, (in Russian). English translation in G. MINTS, ***Selected papers in proof theory***, Bibliopolis, Napoli, 1992.

[82] A. MKRTYCHEV, *Models for the logic of proofs*, ***Logical foundations of Computer Science '97, Yaroslavl'*** (S. Adian and A. Nerode, editors), Lecture Notes in Computer Science, vol. 1234, Springer-Verlag, 1997, pp. 266–275.

[83] R. MONTAGUE, *Syntactical treatments of modality with corollaries on reflection principles and finite axiomatizability*, ***Acta Philosophica Fennica***, vol. 16 (1963), pp. 153–168.

[84] J. MYHILL, *Some remarks on the notion of proof*, ***Journal of Philosophy***, vol. 57 (1960), pp. 461–471.

[85] ———, *Intensional set theory*, ***Intensional mathematics*** (S. Shapiro, editor), North-Holland, 1985, pp. 47–61.

[86] P. S. NOVIKOV, ***Constructive mathematical logic from the viewpoint of the classical one***, Nauka, Moscow, 1977, (in Russian).

[87] I. E. ORLOV, *The calculus of compatibility of propositions*, ***Matematicheskii Sbornik***, vol. 35 (1928), pp. 263–286, (in Russian).

[88] M. PARIGOT, *$\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction*, ***Proceedings of the international conference on logic programming and automated reasoning***, Lecture Notes in Computer Science, vol. 624, Springer-Verlag, 1992, pp. 190–201.

[89] ———, *Strong normalization for second order classical natural deduction*, ***Proceedings of the 8th annual IEEE Symposium on Logic in Computer Science***, IEEE Computer Society Press, 1993, pp. 39–46.

[90] C. PARSONS and W. SIEG, *Introductory note to [44]*, ***Kurt Gödel Collected Works. Volume III*** (S. Feferman, editor), Oxford University Press, 1995, pp. 62–85.

[91] F. PFENNING and H. C. WONG, *On a modal lambda-calculus for* ***S4***, ***Electronic Notes in Computer Science***, vol. 1 (1995).

[92] H. RASIOWA and R. SIKORSKI, ***The mathematics of metamathematics***, Polish Scientific Publishers, 1963.

[93] A. SCEDROV, *Extending Gödel's modal interpretation to type theory and set theory*, ***Intensional mathematics*** (S. Shapiro, editor), North-Holland, 1985, pp. 81–119.

[94] D. SCOTT, *Constructive validity*, ***Symposium on automatic demonstration*** (M. Laudet, D. Lacombe, L. Nolin, and M. Schützenberger, editors), Lecture Notes in Mathematics, vol. 125, Springer-Verlag, Berlin, 1970, pp. 237–275.

[95] S. Shapiro, *Epistemic and intuitionistic arithmetic*, ***Intensional mathematics*** (S. Shapiro, editor), North-Holland, 1985, pp. 11–46.

[96] ———, *Intensional mathematics and constructive mathematics*, ***Intensional mathematics*** (S. Shapiro, editor), North-Holland, 1985, pp. 1–10.

[97] T. Sidon, *Provability logic with operations on proofs*, ***Logical foundations of Computer Science '97, Yaroslavl'*** (S. Adian and A. Nerode, editors), Lecture Notes in Computer Science, vol. 1234, Springer-Verlag, 1997, pp. 342–353.

[98] C. Smoryński, ***Self-reference and modal logic***, Springer-Verlag, Berlin, 1985.

[99] R. Smullyan, ***Diagonalization and self-reference***, Oxford University Press, 1994.

[100] R. Solovay, *Provability interpretations of modal logic*, ***Israel Journal of Mathematics***, vol. 25 (1976), pp. 287–304.

[101] G. Takeuti, ***Proof theory***, North-Holland, 1975.

[102] A. Troelstra, *The scientific work of A. Heyting*, ***Logic and foundations of mathematics*** (D. van Dalen et al., editors), Wolters-Noordhoff Publishing, 1968, pp. 11–46.

[103] ———, *Introductory note to [44]*, ***Kurt Gödel Collected Works. Volume I*** (S. Feferman, editor), Oxford University Press, 1986, pp. 296–299.

[104] A. Troelstra and H. Schwichtenberg, ***Basic proof theory***, Cambridge University Press, Amsterdam, 1996.

[105] A. Troelstra and D. van Dalen, ***Constructivism in mathematics. An introduction***, vol. 1, North-Holland, Amsterdam, 1988.

[106] A.S. Troelstra, *Realizability*, ***Handbook of proof theory*** (S. Buss, editor), Elsevier, 1998, pp. 407–474.

[107] V. Uspensky and V. Plisko, *Intuitionistic logic. commentary on [58] and [59]*, ***Selected works of A. N. Kolmogorov. Volume I: Mathematics and Mechanics*** (V. M. Tikhomirov, editor), Kluwer, 1985, pp. 452–466.

[108] V.A. Uspensky, *Kolmogorov and mathematical logic*, ***The Journal of Symbolic Logic***, vol. 57 (1992), no. 2, pp. 385–412.

[109] J. van Benthem, *Reflections on epistemic logic*, ***Logique & Analyse***, vol. 133–134 (1991), pp. 5–14.

[110] D. van Dalen, *Intuitionistic logic*, ***Handbook of philosophical logic*** (D. Gabbay and F. Guenther, editors), vol. 3, Reidel, 1986, pp. 225–340.

[111] ———, ***Logic and structure***, Springer-Verlag, 1994.

[112] V. Vardanyan, *Arithmetic complexity of predicate logics of provability and their fragments*, ***Soviet Mathematics Doklady***, vol. 33 (1986), pp. 569–572.

[113] A. Visser, *An overview of interpretability logic*, ***Advances in modal logic*** (M. Kracht, M. de Rijke, and H. Wansing, editors), vol. 1, CSLI Publications, Stanford University, 1996, pp. 307–360.

[114] S. Weinstein, *The intended interpretation of intuitionistic logic*, ***Journal of Philosophical Logic***, vol. 12 (1983), pp. 261–270.

[115] T. Yavorskaya (Sidon), *Logics of proofs and provability*, ***Annals of pure and applied logic***, (to appear in 2001), in the volume on the conference ***St. Petersburg Days of Logic and Computability, 1999***, (Yu. Matiyasevich, editor).

[116] R. Yavorsky, *On the logic of the standard proof predicate*, ***Computer Science Logic 2000***, Lecture Notes in Computer Science, vol. 1862, Springer-Verlag, 2000, pp. 527–541.

DEPARTMENT OF COMPUTER SCIENCE
CORNELL UNIVERSITY
ITHACA, NEW YORK 14853, USA

and

DEPARTMENT OF MATHEMATICS
MOSCOW STATE UNIVERSITY
MOSCOW 119899, RUSSIA

E-mail: artemov@cs.cornell.edu