Multi-agent Explicit Knowledge

Tatiana Yavorskaya (Sidon)

Department of Mathematical Logic and Theory of Algorithms,
Faculty of Mechanics and Mathematics,
Moscow State University, Moscow 119992, Russia
tanya@lpcs.math.msu.su

Abstract. Logic of proofs LP, introduced by S. Artemov, originally designed for describing properties of formal proofs, now became a basis for the theory of knowledge with justification. So far, in epistemic systems with justification the corresponding "evidence part", even for multi-agent systems, consisted of a single explicit evidence logic. In this paper we introduce logics describing two interacting explicit evidence systems. We find an appropriate formalization of the intended semantics and prove the completeness of these logics with respect to both symbolic and arithmetical models. Also, we find the forgetful projections for the logics with two proof predicates which are extensions of the bimodal logic $\mathsf{S4}^2$.

1 Introduction

The Logic of Proofs LP introduced by S. Artemov in 1995 (see the detailed description in [1, 2]) was originally designed to express in logic the notion of a proof. It is formulated in the propositional language enriched by new atoms $[\![t]\!]F$ with the intended meaning *"t is a proof of F"*. Proofs are represented by *proof terms* constructed from *proof variables* and *proof constants* by means of three elementary computable operations: binary $\cdot$, $+$ and unary $!$ specified by the axioms

$$[\![t]\!](A \to B) \to ([\![s]\!]A \to [\![t \cdot s]\!]B) \qquad \textit{application}$$
$$[\![t]\!]A \to [\![t + s]\!]A, \quad [\![s]\!]A \to [\![t + s]\!]A \qquad \textit{nondeterministic choice}$$
$$[\![t]\!]A \to [\![!t]\!][\![t]\!]A \qquad \textit{positive proof checker}$$

LP is axiomatized over propositional calculus by the above axioms and the principle

$$[\![t]\!]A \to A \qquad \textit{weak reflexivity}$$

The rules of inference are *modus ponens* and *axiom necessitation rule*. The latter allows to specify proof constants as proofs of the concrete axioms

$$\frac{}{[\![a]\!]\mathsf{A}}, \quad \text{where } a \text{ is an axiom constant, } \mathsf{A} \text{ is an axiom of LP.}$$

The intended semantics for LP is given by formal proofs in Peano Arithnmetic PA: proof variables are interpreted by codes of PA-derivations, $[\![t]\!]F$ stands for

D. Grigoriev, J. Harrison, and E.A. Hirsch (Eds.): CSR 2006, LNCS 3967, pp. 369–380, 2006.

the arithmetical proof predicate "t is a proof of F". It is proven in [2] that LP is arithmetically complete with respect to the class of all proof systems. Furthermore, LP suffices to realize Gödel's provability logic S4 and thus provides S4 and intuitionistic logic with the exact provability semantics.

In [3] it was suggested to treat $[\![t]\!]F$ as a new type of knowledge operator called *evidence–based knowledge* with the meaning "t is an evidence for F." Evidence based knowledge (EBK) systems are obtained by augmenting a multi–agent logic of knowledge with a system of evidence assertions $[\![t]\!]F$. Three main cases of EBK–systems were introduced in [3] in which the base knowledge logic is T_n, $\mathsf{S4}_n$ or $\mathsf{S5}_n$. The evidence part for all of them consists of a single logic of proofs LP.

In this paper we study multiple interacting EBK–systems, namely, we study logics that describe the behavior of two reasoning agents $\mathcal{P}_1$ and $\mathcal{P}_2$ which somehow communicate to each other. For simplicity, let us think about a reasoning agent as a proof system, then evidences are proofs in this system. We develop a language with two proof operators $[\![\cdot]\!]_1(\cdot)$ and $[\![\cdot]\!]_2(\cdot)$ representing proof predicates for $\mathcal{P}_1$ and $\mathcal{P}_2$. In general, proofs of these two systems are distinct, so proof terms for a proof system $\mathcal{P}_i$ $(i = 1, 2)$ are constructed from its own atomic proofs represented by proof variables p_k^i and proof constants c_k^i. We suppose that both $\mathcal{P}_1$ and $\mathcal{P}_2$ has all the power of LP, so we reserve a copy of LP–operations $\times_i$, $+_i$ and $!_i$ for application, nondeterministic choice and positive proof checker in $\mathcal{P}_i$ $(i = 1, 2)$.

For the minimal logic of two proof systems denoted by LP^2 we assume that there is no communication between them, except that all axioms are common knowledge, so we extend the axiom necessitation rule and allow it to derive all the formulas

$$[\![c_{j_1}^{k_1}]\!]_{k_1}[\![c_{j_2}^{k_2}]\!]_{k_2}\ldots[\![c_{j_n}^{k_n}]\!]_{k_n}A, \text{ where all } k_i \in \{1,2\},\ A \text{ is an axiom.}$$

Going further, we may assume that the two systems $\mathcal{P}_1$ and $\mathcal{P}_2$ are allowed to communicate, that is, one of the proof systems is able to derive something about the other one. We study two types of communications.

Proof checking. We assume that $\mathcal{P}_2$ can verify all proofs of $\mathcal{P}_1$ and introduce a unary operation $!_1^2$ specified by the axiom

$$[\![t]\!]_1A \to [\![!_1^2t]\!]_2[\![t]\!]_1A.$$

Further, we can consider the case when both $\mathcal{P}_1$ and $\mathcal{P}_2$ are able to verify each other, then we add the dual operation $!_2^1$ with the specification

$$[\![t]\!]_2A \to [\![!_2^1t]\!]_1[\![t]\!]_2A.$$

The resulting logics are denoted by $\mathsf{LP}_!^2$ and $\mathsf{LP}_{!!}^2$ respectively.

Proof embedding. Here we suppose that all proofs of $\mathcal{P}_1$ can be converted to $\mathcal{P}_2$–proofs; this is done by the operation $\uparrow_1^2$ specified by the principle

$$[\![t]\!]_1A \to [\![\uparrow_1^2 t]\!]_2A.$$

If $\mathcal{P}_1$ can also imitate $\mathcal{P}_2$–proof, we add a converse operation $\uparrow_2^1$ with the specification

$$[\![t]\!]_2 A \to [\![\uparrow_2^1 t]\!]_1 A.$$

We denote the resulting logics by $\mathsf{LP}^2_{\uparrow}$ and $\mathsf{LP}^2_{\uparrow\uparrow}$.

In this paper for all the logics L mentioned above we do the following:

- describe symbolic semantics and prove completeness of L;
- find the forgetful projection of L, i.e. a bimodal logic obtained from L by replacing all occurrences of $[\![t]\!]_i$ by $\Box_i$ for $i = 1, 2$;
- describe arithmetical interpretation and prove completeness of L.

The structure of the paper is the following. In section 2 we give a precise description of the language and the logics we are dealing with. In section 3 the modal counterparts of all the described logics are found. It turned out that the forgetful projections of $\mathsf{LP}^2_!$ and $\mathsf{LP}^2_{!!}$ coincide with the projections of $\mathsf{LP}^2_{\uparrow}$ and $\mathsf{LP}^2_{\uparrow\uparrow}$ respectively. Section 4 is devoted to symbolic and arithmetical semantics.

2 Explicit Evidence Logics for Two Agents: Definitions

Definition 1. *The minimal language L of the bimodal explicit evidence logic is denoted by LP^2. It contains*

- *propositional variables $SVar = \{S_1, S_2, \ldots\}$;*
- *two disjoint sets of proof variables $PVar^i = \{p_1^i, p_2^i, \ldots\}$ and two disjoint sets of proof constants $\{c_1^i, c_2^i, \ldots\}$ where $i = 1, 2$;*
- *two copies of every operation on proofs from LP: binary $\times_1$, $+_1$, $\times_2$, $+_2$ and unary $!_1$ and $!_2$;*
- *Boolean connectives and two operational symbols $[\![\cdot]\!]_1(\cdot)$ and $[\![\cdot]\!]_2(\cdot)$ of the type* proof $\to$ (proposition $\to$ proposition)

We also consider extensions of L. The first option is to add one or both of the unary functional symbols $!_1^2$ and $!_2^1$; we denote the result by $\mathsf{LP}^2_!$, $\mathsf{LP}^2_{!!}$ respectively. Another option is to add one or both of the unary functional symbols $\uparrow_1^2$ and $\uparrow_2^1$; the result is denoted by $\mathsf{LP}^2_{\uparrow}$, $\mathsf{LP}^2_{\uparrow\uparrow}$ respectively.

For every language L from the definition above we define two sets of terms $Tm_i(L)$, $(i = 1, 2)$. For $L = \mathsf{LP}^2$ the set $Tm_i(L)$ consists of all terms constructed from variables and constants labelled with sup-i by operations labelled by i. Namely, for $i = 1, 2$, every proof variable p_j^i or proof constant c_j^i is an element of $Tm_i(L)$ and if $t, s \in Tm_i(L)$, then $t \times_i s$, $t +_i s$ and $!_i t$ belong to $Tm_i(L)$ too. For the extensions of the minimal language we add the following clauses to the definition of terms

- for $L = \mathsf{LP}^2_!$, if $t \in Tm_1(L)$ then $!_1^2 t \in Tm_2(L)$;
- for $L = \mathsf{LP}^2_{!!}$, if $t \in Tm_1(L)$ then $!_1^2 t \in Tm_2(L)$ and if $t \in Tm_2(L)$ then $!_2^1 t \in Tm_1(L)$;

- for $L = \mathsf{LP}^2_{\uparrow}$, if $t \in Tm_1(L)$ then $\uparrow^2_1 t \in Tm_2(L)$;
- for $L = \mathsf{LP}^2_{\uparrow\uparrow}$, if $t \in Tm_1(L)$ then $\uparrow^2_1 t \in Tm_2(L)$ and if $t \in Tm_2(L)$ then $\uparrow^1_2 t \in Tm_1(L)$.

Formulas of the language L are constructed from sentence variables by boolean connectives and according to the rule: for $i = 1, 2$ if $t \in Tm_i(L)$ and F is a formula of L then $[\![t]\!]_i F$ is a formula of L too. The set of all formulas is denoted by $Fm(L)$. Formulas of the form and $[\![t]\!]_i F$ are called *q-atomic*, the set of such formulas is denoted by $QFm_i(L)$. We write $QFm(L)$ for $QFm_1(L) \cup QFm_2(L)$.

Operations on proofs are specified by the following formulas (t, s are terms, A, B are formulas):

$$\begin{array}{ll}
\mathrm{Ax}(\times_i) & [\![t]\!]_i(A \to B) \to ([\![s]\!]_i A \to [\![t \times_i s]\!]_i B) \\
\mathrm{Ax}(+_i) & [\![t]\!]_i A \to [\![t +_i s]\!]_i A, \quad [\![s]\!]_i A \to [\![t +_i s]\!]_i A \\
\mathrm{Ax}(!_i) & [\![t]\!]_i A \to [\![!_i t]\!]_i [\![t]\!]_i A \\
\mathrm{Ax}(!^2_1) & [\![t]\!]_1 A \to [\![!^2_1 t]\!]_2 [\![t]\!]_1 A \\
\mathrm{Ax}(!^1_2) & [\![t]\!]_2 A \to [\![!^1_2 t]\!]_1 [\![t]\!]_2 A \\
\mathrm{Ax}(\uparrow^2_1) & [\![t]\!]_1 A \to [\![\uparrow^2_1 t]\!]_2 A \\
\mathrm{Ax}(\uparrow^1_2) & [\![t]\!]_2 A \to [\![\uparrow^1_2 t]\!]_1 A
\end{array}$$

Definition 2. *For every language L from Definition 1 we define the corresponding bimodal logic of proofs L. It is axiomatized by the following schemas:*

A0 *classical propositional axioms*
A1 $[\![t]\!]_i A \to A, \quad i = 1, 2$
A2... *axioms for all operations of L.*

The rules of inference are modus ponens and axiom necessitation rule

$$[\![c^{k_1}_{j_1}]\!]_{k_1} [\![c^{k_2}_{j_2}]\!]_{k_2} \ldots [\![c^{k_n}_{j_n}]\!]_{k_n} A, \text{ where all } k_i \in \{1, 2\}, \ A \text{ is an axiom.}$$

Informally speaking, the language LP^2 describes the structure which contains objects of three types: *propositions* represented by formulas, *proofs*$_1$ and *proofs*$_2$ represented by proof terms. We suppose that there are two proof systems $\mathcal{P}_1$ and $\mathcal{P}_2$; the system $\mathcal{P}_i$ tries to find $t \in$ *proofs*$_i$ for $A \in$ *propositions*. The structure is supplied with two proof predicates $[\![t]\!]_1 A$ and $[\![t]\!]_2 A$, which correspond to $\mathcal{P}_1$ and $\mathcal{P}_2$. Both proof predicates are supposed to be recursive. For every $p \in$ *proofs*$_i$ the set of propositions proved by p in $\mathcal{P}_i$ is finite and the function that maps proofs to the corresponding sets is total recursive.

Both proof systems $\mathcal{P}_1$ and $\mathcal{P}_2$ are supplied with operations on proofs taken from LP, thus, they are capable of internalizing there own proofs. The minimal language LP^2 corresponds to the situation when two proof systems do not communicate. The only information about $\mathcal{P}_1$ which is available to $\mathcal{P}_2$ and vise versa is transferred via the axiom necessitation rule. For example, the second proof system knows that $!_1$ is a proof checker of the first one since we can derive $[\![c^2]\!]_2([\![t]\!]_1 A \to [\![!t]\!]_1 [\![t]\!]_1 A)$. Externally we can prove that something is provable in $\mathcal{P}_1$ iff it is provable in $\mathcal{P}_2$, that is, the following two assertions are equivalent:

$$\text{there exists a term } t \in Tm_1(\mathsf{LP}^2) \text{ such that } \mathsf{LP}^2 \vdash [\![t]\!]_1 A$$
$$\text{and}$$
$$\text{there exists a term } s \in Tm_2(\mathsf{LP}^2) \text{ such that } \mathsf{LP}^2 \vdash [\![s]\!]_2 A$$

However, this fact cannot be derived in LP^2, that is, there is no term $t \in Tm_2(\mathsf{LP}^2)$ such that $\mathsf{LP}^2 \vdash [\![p^1]\!]_1 S \to [\![t]\!]_2 S$ (this fact easily can be proven using symbolic semantics from section 4). So, neither $\mathcal{P}_1$ nor $\mathcal{P}_2$ is able to formalize or proof the equivalence just mentioned.

The communication between the two proof systems becomes possible in the extensions of LP^2. In $\mathsf{LP}^2_!$ and $\mathsf{LP}^2_\uparrow$ it is one-way: $\mathcal{P}_2$ can derive some facts about $\mathcal{P}_1$. In $\mathsf{LP}^2_{!!}$ in $\mathsf{LP}^2_{\uparrow\uparrow}$ information can be transferred symmetrically both-ways. Operations $!^2_1$ and $!^1_2$ are proof checkers. $\mathsf{LP}^2_!$ corresponds to the case when $\mathcal{P}_2$ is able to check proofs of $\mathcal{P}_1$; in $\mathsf{LP}^2_{!!}$ we suppose that both of $\mathcal{P}_i$ can proof-check each other. Operations $\uparrow^2_1$ or $\uparrow^1_2$ appear if one of the systems can prove everything that the other one can.

Operations $!^2_1$ and $\uparrow^2_1$ can imitate each other in the following sense.

Lemma 1. *1. For every term $t \in Tm_1(\mathsf{LP}^2_!)$ and formula $F \in Fm(\mathsf{LP}^2_!)$, there is a term $s \in Tm_2(\mathsf{LP}^2_!)$ such that $\mathsf{LP}^2_! \vdash [\![t]\!]_1 F \to [\![s]\!]_2 F$.*

2. For every term $t \in Tm_1(\mathsf{LP}^2_\uparrow)$ and formula $F \in Fm(\mathsf{LP}^2_\uparrow)$, there is a term $s \in Tm_2(\mathsf{LP}^2_\uparrow 0$ such that $\mathsf{LP}^2_\uparrow \vdash [\![t]\!]_1 F \to [\![s]\!]_2 [\![t]\!]_1 F$.

Proof. 1. Derive in $\mathsf{LP}^2_!$

$[\![t]\!]_1 F \to [\![!^2_1 t]\!]_2 [\![t]\!]_1 F$
$[\![c^2]\!]_2([\![t]\!]_1 F \to F)$
$[\![t]\!]_1 F \to [\![c^2 \times_2 (!^2_1 t)]\!]_2 F$
take $s = c^2 \times_2 (!^2_1 t)$

2. Derive in $\mathsf{LP}^2_\uparrow$

$[\![t]\!]_1 F \to [\![!_1 t]\!]_1 [\![t]\!]_1 F$
$[\![!_1 t]\!]_1 [\![t]\!]_1 F \to [\![\uparrow^2_1 !_1 t]\!]_2 [\![t]\!]_1 F$
$[\![t]\!]_1 F \to [\![\uparrow^2_1 (!_1 t)]\!]_2 [\![t]\!]_1 F$
take $s = \uparrow^2_1 (!_1 t)$.

Lemma 2 (Internalization property). *Let L be one of the logics from Definition 2. If $L \vdash F$, then for $i = 1, 2$ there exists a term t_i constructed from constants with the help of operations $\times_i$ and $!_i$ such that $L \vdash [\![t_i]\!]_i F$.*

Proof. Standard induction on derivation of F.

Lemma 3. *Let L be one of the logics from definition 2. For $i = 1, 2$ let δ_i be a $\wedge,\vee$–combination of q-atoms from $QFm_i(L)$. Let δ stand for a $\wedge,\vee$–combination of q-atoms from $QFm_1(L) \cup QFm_2(L)$.*

1. There exists a term t_i such that $L \vdash \delta_i \to [\![t_i]\!]_i \delta_i$.

2. If L contains either $!^2_1$ or $\uparrow^2_1$ then there exists a term $t \in Tm_2(L)$ such that $L \vdash \delta \to [\![t]\!]_2 \delta$.

3. If L contains either $!^1_2$ or $\uparrow^1_2$ then there exists a term $t \in Tm_1(L)$ such that $L \vdash \delta \to [\![t]\!]_1 \delta$.

Proof. 1. Induction on the construction of δ_i. If $\delta_i = [\![t]\!]_i F$ then apply $\mathrm{Ax}(!_i)$ to obtain $L \vdash \delta_i \to [\![!_i t]\!]_i \delta_i$. If $\delta_i = \alpha_i \wedge \beta_i$ or $\delta_i = \alpha_i \vee \beta_i$ then, by the induction hypothesis, there exist terms u and v such that $L \vdash \alpha_i \to [\![u]\!]_i \alpha_i$ and $L \vdash \beta_i \to$

$[\![v]\!]_i\beta_i$. By the axiom necessitation rule, $L \vdash [\![c^i]\!]_i(\alpha_i \to (\beta_i \to (\alpha_i \wedge \beta_i))$. Using $\mathrm{Ax}(\times_i)$, we derive

$$L \vdash \alpha_i \wedge \beta_i \to [\![c^i \times_i u \times_i v]\!]_i(\alpha_i \wedge \beta_i).$$

By axiom necessitation we also have $L \vdash [\![c_1^i]\!]_i(\alpha_i \to \alpha_i \vee \beta_i)$ and $L \vdash [\![c_2^i]\!]_i(\beta_i \to \alpha_i \vee \beta_i)$. Hence $L \vdash \alpha_i \to [\![c_1^i \times_i u]\!]_i(\alpha_i \vee \beta_i)$ and $L \vdash \beta_i \to [\![c_2^i \times_i v]\!]_i(\alpha_i \vee \beta_i)$. Therefore $L \vdash \alpha_i \vee \beta_i \to [\![(c_1^i \times_i u) +_i (c_2^i \times_i v)]\!]_i(\alpha_i \vee \beta_i)$.

2. The induction step is similar to the previous case. For the induction base now we should consider two options $[\![t]\!]_1 F$ and $[\![s]\!]_2 F$ instead of one. The second option is treated similarly with the previous case. For $\delta = [\![t]\!]_1 F$ we have $\mathsf{LP}_!^2 \vdash [\![t]\!]_1 F \to [\![!_1^2 t]\!]_2 [\![t]\!]_1 F$. In $\mathsf{LP}_\uparrow^2$ we reason as follows: $\mathsf{LP}_\uparrow^2 \vdash [\![t]\!]_1 F \to [\![!_1 t]\!]_1 [\![t]\!]_1 F$ and $\mathsf{LP}_\uparrow^2 \vdash [\![!_1 t]\!]_1 \delta_{1,2} \to [\![\uparrow_1^2 !_1 t]\!]\delta_{1,2}$. Hence $\mathsf{LP}_\uparrow^2 \vdash \delta_{1,2} \to [\![\uparrow_1^2 !_1 t]\!]\delta_{1,2}$.

3. Similar to 2.

3 Realization of Bimodal Logics

In [2] it is proven that LP is able to realize all derivations in the modal logic $\mathsf{S4}$, namely, if A is a theorem of $\mathsf{S4}$ then there is an assignment of LP–terms to all occurrences of $\Box$'s in A such that the resulting formula is a theorem in LP. In this section we describe the modal counterparts of the logics LP^2, $\mathsf{LP}_!^2$ and $\mathsf{LP}_\uparrow^2$.

We need the bimodal logic $\mathsf{S4}^2$ and its extension $\mathsf{S4}^2_{\mathsf{mon}}$. $\mathsf{S4}^2$ is given by the following axioms and rules of inference: for $i = 1, 2$,

A1 propositional tautologies
A2 $\Box_i A \to A$
A3 $\Box_i(A \to B) \to (\Box_i A \to \Box_i B)$
A4 $\Box_i A \to \Box_i \Box_i A$
R1 Modus Ponens: $A, A \to B \vdash B$
R2 Necessitation: if $\vdash A$ then $\vdash \Box_i A$.

$\mathsf{S4}^2_{\mathsf{mon}}$ is an extension of $\mathsf{S4}^2$ by the principle

A5 $\Box_1 F \to \Box_2 F$.

We prove that the analog of the realization theorem for $\mathsf{S4}$ and LP holds for the following pairs of logics: $\mathsf{S4}^2$ and LP^2, $\mathsf{S4}^2_{\mathsf{mon}}$ and $\mathsf{LP}_!^2$, $\mathsf{S4}^2_{\mathsf{mon}}$ and $\mathsf{LP}_\uparrow^2$. We need the following definition.

Definition 3. *Let L be one of the languages from definition 1. Suppose that A is a formula with two modalities. A realization of A in the language L is a formula $A^r \in Fm(L)$ which is obtained from A by substitution of terms from $Tm_i(L)$ for all occurrences of $\Box_i$ in A. A realization is normal if all negative occurrences of modalities are assigned proof variables.*

Theorem 1. *1. $\mathsf{S4}^2 \vdash A$ iff there exists a normal realization r in the language LP^2 such that $\mathsf{LP}^2 \vdash A^r$.*

2. For $L \in \{\mathsf{LP}_!^2, \mathsf{LP}_\uparrow^2\}$, $\mathsf{S4}^2_{\mathsf{mon}} \vdash A$ iff there exists a normal realization r in the language L such that $L \vdash A^r$.

The proof of this theorem goes along the lines of the proof of realization of $\mathsf{S4}$ in LP (sf. [2]). First of all, we need the normalized Gentzen-style versions of $\mathsf{S4}^2$ and $\mathsf{S4}^2_{\mathsf{mon}}$. Sequential calculus for $\mathsf{S4}^2$, denoted by $\mathsf{GS4}^2$, has the same axioms and rules as sequential calculus for classical propositional logic plus four modal rules (two for each modality):

$$(\text{Left}\Box_i)\ \frac{A, \Gamma \Rightarrow \Delta}{\Box_i A, \Gamma \Rightarrow \Delta} \qquad (\text{Right}\Box_i)\ \frac{\Box_i \Gamma \Rightarrow A}{\Box_i \Gamma \Rightarrow \Box_i A} \qquad (i = 1, 2).$$

In the Gentzen-style version of $\mathsf{S4}^2_{\mathsf{mon}}$ denoted by $\mathsf{GS4}^2_{\mathsf{mon}}$ the rule $(\text{Right}\Box_2)$ is replaced by a stronger version

$$\frac{\Box_1 \Gamma_1, \Box_2 \Gamma_2 \Rightarrow A}{\Box_1 \Gamma_1, \Box_2 \Gamma_2 \Rightarrow \Box_2 A}.$$

Theorem 2. *For a logic $L \in \{\mathsf{S4}^2, \mathsf{S4}^2_{\mathsf{mon}}\}$ the following connection between L and its Gentzen-style version $\mathcal{G}$ holds:*

$$\mathcal{G} \vdash \Gamma \Rightarrow \Delta \text{ iff } L \vdash \bigwedge \Gamma \to \bigvee \Delta.$$

Theorem 3. *Any logic $\mathcal{G} \in \{GS4^2, G\mathsf{S4}^2_{\mathsf{mon}}\}$ enjoys cut-elimination: if $\mathcal{G} \vdash \Gamma \Rightarrow \Delta$ then $\Gamma \Rightarrow \Delta$ can be derived in $\mathcal{G}$ without using of the Cut-rule.*

Lemma 4. *1. $\mathsf{GS4}^2 \vdash \Gamma \Rightarrow \Delta$ iff there exists a normal realization r such that $\mathsf{LP}^2 \vdash (\bigwedge \Gamma \to \bigvee \Delta)^r$.*

2. For $L \in \{\mathsf{LP}^2_!, \mathsf{LP}^2_\uparrow\}$, $\mathsf{GS4}^2_{\mathsf{mon}} \vdash \Gamma \Rightarrow \Delta$ iff there exists a normal realization r in the language L such that $L \vdash (\bigwedge \Gamma \to \bigvee \Delta)^r$.

Proof. Similar to the proof of the realization theorem for LP. Goes by induction on the cut-free proof of $\Gamma \Rightarrow \Delta$. Uses Internalization and δ-completeness.

4 Symbolic and Arithmetical Semantics

Models of multi-agent logics of explicit knowledge below are natural generalizations of Mkrtychev models for LP (cf. [6]).

Definition 4. *Let L be any language from definition 1. An L–model $\mathcal{M} = (\#, v)$ consists of two objects*

- *$\#$ is a mapping from proof terms of L to sets of formulas of L, called an evidence function;*
- *v is a truth evaluation of sentence variables.*

For every functional symbol from L the evidence function $\#$ should satisfy the corresponding closure condition from the list given below: suppose that t, s are in $Tm_i(L)$, $i = 1, 2$

- if $(A \to B) \in \#(t)$, $A \in \#(s)$ then $B \in \#(t \times_i s)$;
- if $A \in \#(t)$ then $A \in \#(t +_i s)$ and $A \in \#(s +_i t)$;
- if $A \in \#(t)$ then $[\![t]\!]_i A \in \#(!_i t)$;
- if $A \in \#(u)$ and $u \in Tm_1(L)$ then $[\![u]\!]_1 A \in \#(!_1^2 u)$;
- if $A \in \#(v)$ and $v \in Tm_2(L)$ then $[\![v]\!]_2 A \in \#(!_2^1 v)$;
- if $A \in \#(u)$ and $u \in Tm_1(L)$ then $A \in \#(\uparrow_1^2 u)$;
- if $A \in \#(v)$ and $v \in Tm_2(L)$ then $A \in \#(\uparrow_2^1 v)$.

Definition of the truth relation $\mathcal{M} \models A$ is inductive: for propositional variables $\mathcal{M} \models S$ iff $v(S) = true$, $\models$ commutes with Boolean connectives and for $t_i \in Tm_i$

$$\mathcal{M} \models [\![t]\!]_i A \ \rightleftharpoons \ A \in \#(t) \text{ and } \mathcal{M} \models A.$$

A model $\mathcal{M} = (\#, v)$ is called *finitely generated* (or f.g. for short) if

- for every term t the set $\#(t)$ is finite; the set $\{p \in PVar |\ \#(p) \neq \emptyset\}$ is finite;
- the set of terms, for which the converse of the conditions on the evidence function does not hold, is finite;
- the set $\{S \in SVar \mid v(S) = true\}$ is finite.

Definition 5. *For any logic L from definition 2 a constant specification CS is any finite set of formulas derived by the axiom necessitation rule. We say that $L \vdash A$ meeting CS if all axiom necessitation rules in the derivation of A introduce formulas from CS. We say that an L–model $\mathcal{M}$ meets CS if $\mathcal{M} \models (\bigwedge CS)$.*

Theorem 4. *Let L be any logic from definition 2.*

1. If $L \vdash A$ meeting CS then for every L-model $\mathcal{M}$ meeting CS one has $\mathcal{M} \models A$.

2. If $L \nvdash A$ meeting CS then there exists a f.g. L-model $\mathcal{M}$ meeting CS such that $\mathcal{M} \not\models A$.

Proof. We give the sketch of the proof for $L = \mathsf{LP}^2$; for the remaining systems the proof differs in saturation and completion algorithms (see below) to which the cases corresponding to the additional operations should be added. It is enough to consider the case $CS = \emptyset$; the general case can be reduces to this one by the deduction theorem which holds in all logics L. We omit all the proofs of technical lemmas.

Soundness can be easily proven by induction on the derivation of A. In order to prove completeness suppose that $\mathsf{LP}^2 \nvdash A$. We will construct a finitely generated model $\mathcal{M} = (\#, v)$ such that $\mathcal{M} \not\models A$.

Step 1: Saturation algorithm. It constructs a finite set of formulas $Sat(A)$ which is called an *adequate set*. We need the following definition: the *complexity of a proof term t* denoted by $|t|$ is the length of the longest branch in the tree representing this term. The saturation algorithm works as follows:

1. Initialization. Put $Sat_0(A) := SubFm(A)$. Calculate the maximal complexity of terms which occur in A; let N denote the result.

2. For every $l = 1, \dots, N+1$ we calculate the set $Sat_l(A)$ as follows.
   - Initially $Sat_l(A) := Sat_{l-1}(A)$.
   - if $[\![t]\!]_i(A \to B), [\![s]\!]_i A \in Sat_{l-1}(A)$ then extend $Sat_l(A)$ by $[\![t \times_i s]\!]_i B$;
   - if $[\![t]\!]_i A \in Sat_{l-1}(A)$ and $|s| \le l$ then extend $Sat_l(A)$ by $[\![t +_i s]\!]_i A$ and $[\![s +_i t]\!]_i A$;
   - if $[\![t]\!]_i A \in Sat_{l-1}(A)$ then extend $Sat_l(A)$ by $[\![!_i t]\!]_i [\![t]\!]_i A$.
3. Put $Sat(A) := Sat_{N+1}(A)$.

Lemma 5. *(Properties of adequate sets.) For every $l = 0, \dots, N+1$,*

1. *$Sat_l(A)$ is closed under subformulas, that is, $SubFm(Sat_l(A)) \subseteq Sat_l(A)$.*
2. *If $G \in Sat_{l+1}(A) \setminus Sat_l(A)$ then G has the form $[\![t]\!]E$ and $|t| \ge l+1$.*
3. *If $[\![t]\!]_i(F \to G), [\![s]\!]_i F \in Sat(A)$ and $|t \times_i s| \le N$ then $[\![t \times_i s]\!]_i G \in Sat(A)$.*
   If $[\![t]\!]_i G \in Sat(A)$ and $|t +_i s| \le N$, then $[\![t +_i s]\!]_i G, [\![s +_i t]\!]_i G \in Sat(A)$.
   If $[\![t]\!]_i G \in Sat(A)$ and $|!_i t| \le N$ then $[\![!_i t]\!]_i [\![t]\!]_i G \in Sat(A)$.

Proof. Joint induction on l.

Step 2. Now we describe a translation of the language LP^2 into the pure propositional language. For every q-atom $[\![t]\!]_i B \in Sat(A)$ we reserve a fresh propositional variable $S_{t,i,B}$. For every formula G whose all q-atomic subformulas belong to $Sat(A)$ by G' we denote the result of substitution of all outermost occurrences of q-atomic subformulas in G by the corresponding propositional variables. Namely, we define G' by induction on the construction of G in the following way: for propositional variables $S' \rightleftharpoons S$; $(\cdot)'$ commutes with boolean connectives and $([\![t]\!]_i B)' \rightleftharpoons S_{t,i,B}$.

Let $\mathrm{Ax}(A)$ stand for the conjunction of all substitutional instances of axioms A1–A4 whose all q-atomic subformulas are from $Sat(A)$. Put $A_p \rightleftharpoons (\mathrm{Ax}(A) \to A)'$. Since $\mathsf{LP}^2 \nvdash A$ we conclude that A_p is not provable in propositional logic (otherwise after the reverse substitution of $[\![t]\!]_i B$ for $S_{t,i,B}$ in the derivation of A_p in propositional calculus we get $\mathsf{LP}^2 \vdash \mathrm{Ax}(A) \to A$, hence $\mathsf{LP}^2 \vdash A$). Therefore, there exists an evaluation w of propositional letters from A_p by (*true*, *false*) such that $w(A_p) = \mathit{false}$. Define

$$\begin{aligned} \Gamma_0 &\rightleftharpoons \{B \in Sat(A) \mid w(B') = \mathit{true}\}, \\ \Delta_0 &\rightleftharpoons \{B \in Sat(A) \mid w(B') = \mathit{false}\}. \end{aligned}$$

Lemma 6. *The sets Γ_0 and Δ_0 has the following properties:*

1. *$\Gamma_0 \cap \Delta_0 = \emptyset$.*
2. *If $[\![t]\!]E \in \Gamma_0$ then $E \in \Gamma_0$.*
3. *If $[\![t]\!]_i(F \to G)$, $[\![s]\!]_i F \in \Gamma_0$ and $|t \times_i s| \le N$ then $[\![t \times_i s]\!]_i G \in \Gamma_0$.*
   If $[\![t]\!]_i G \in \Gamma_0$ and $|t +_i s| \le N$ then $[\![t +_i s]\!]_i G \in \Gamma_0$ and $[\![s +_i t]\!]_i G \in \Gamma_0$.
   If $[\![t]\!]_i G \in \Gamma_0$ and $|!_i t| \le N$ then $[\![!_i t]\!]_i [\![t]\!]_i G \in \Gamma_0$.

Step 3. Completion algorithm. It goes through infinite number of iterations; the l-th iteration produces the set Γ_l which is finite. Start with Γ_0. For every $l = 1, 2, \dots$ on the l-th iteration construct the set Γ_l as follows

– Initially $\Gamma_l := \Gamma_{l-1}$.
– if $[\![t]\!]_i(A \to B), [\![s]\!]_i A \in \Gamma_{l-1}$ then extend Γ_l by $[\![t \times_i s]\!]B$;
– if $[\![t]\!]_i A \in \Gamma_{l-1}$ and $|s| \le l$ then extend Γ_l by $[\![t +_i s]\!]_i A$ and $[\![s +_i t]\!]_i A$;
– if $[\![t]\!]_i A \in \Gamma_{l-1}$ then extend Γ_l by $[\![!_i t]\!]_i [\![t]\!]_l A$;
– Go to the next l.

Put $\Gamma := \bigcup_l \Gamma_l$.

Lemma 7. *For every $l = 0, 1, 2, \ldots$,*

1. *The set Γ_l is finite and $\Gamma_l \cap \Delta = \emptyset$*
2. *If $E \in \Gamma_{l+1} \setminus \Gamma_l$ then E is of the form $[\![t]\!]G$ and $|t| \ge N + l + 1$.*
3. *$\Gamma_l \cup \Delta_0$ is closed under subformulas, that is, $SubFm(\Gamma_l \cup \Delta) \subseteq \Gamma_l \cup \Delta$.*
4. *If $[\![t]\!]_i(F \to G), [\![s]\!]_i F \in \Gamma$ then $[\![t \times_i s]\!]_i G \in \Gamma$. If $[\![t]\!]_i G \in \Gamma$ then $[\![t +_i s]\!]_i G \in \Gamma$ and $[\![s +_i t]\!]_i G \in \Gamma$. If $[\![t]\!]_i G \in \Gamma$ then $[\![!_i t]\!]_i [\![t]\!]_i G \in \Gamma$.*
5. *For every term t the set $I(t) = \{E \mid [\![t]\!]E \in \Gamma\}$ is finite and the function $t \mapsto I(t)$ is primitive recursive.*

Proof. Induction on l.

Step 4. For every $t \in Tm_i$ and $S \in SVar$ put

$$\#(t) \rightleftharpoons \{E \mid [\![t]\!]E \in \Gamma\} \qquad\qquad v(S) \rightleftharpoons w(S).$$

Lemma 8. *For every formula G one has*

$$G \in \Gamma \Rightarrow \mathcal{M} \models G;$$
$$G \in \Delta \Rightarrow \mathcal{M} \not\models G.$$

Proof. Induction on G. We use lemma 7.

From lemmas 8 and 7 it follows that $\mathcal{M}$ is a finitely generated model for LP^2. Since $w(A') = \mathit{false}$ we conclude $A \in \Delta$, hence $\mathcal{M} \not\models A$. This completes the proof of the theorem.

Corollary 1. LP^2 *is decidable.*

Epistemic semantics for LP^2 is given by the following natural generalization of Fitting models (cf. [4]). For any language L from definition 1 one could define a *Fitting model* as follows. An *L–model* $\mathcal{M} = (W, R_1, R_2, \mathcal{E}, v)$ has the following parameters

– a nonempty set of possible worlds W;
– two reflexive transitive accessibility relations on W denoted by R_1, R_2
– an evidence function $\mathcal{E}$ which maps $W \times Tm(L)$ to sets of formulas of L,
– for every $x \in W$ a truth evaluation $v(x)$ maps propositional variables to $\{\mathit{true}, \mathit{false}\}$.

We require that for every node $x \in W$ the restriction of $\mathcal{E}$ to x satisfies all the conditions for # and $\mathcal{E}$ is monotone in the following sense: for $i = 1, 2$ if xR_iy and $t \in Tm_i(L)$ then $\mathcal{E}(x,t) \subseteq \mathcal{E}(y,t)$.

The truth relation for every node $x \in W$ is defined in the standard way; we put $\mathcal{M}, x \models [\![t]\!]_i F$ iff $F \in \mathcal{E}(x,t)$ and $\mathcal{M}, y \models F$ for every $y \in W$ such that xR_iy.

Note that a model in the sense of definition 4 is a Fitting model, namely, take W a singleton set and R_1, R_2 total relations on W. It is easy to prove that all the logics considered in this paper are sound and complete with respect to the models just described. In particular, the completeness with respect to Fitting semantics follows from Theorem 4 and the fact that aforementioned Mkrychev models are singleton versions of the corresponding Fitting models.

Now let us describe the interpretation of bimodal logics of proofs in Peano Arithmetic PA (the definition of PA and related topics can be found in [7]).

Definition 6. *A normal proof predicate Prf is an arithmetical provably Δ_1 formula satisfying the following conditions:*

1) for every arithmetical formula φ $\mathsf{PA} \vdash \varphi$ iff there exists a natural number n such that $Prf(n, \ulcorner\varphi\urcorner)$;

2) for every n the set $Th(n) \rightleftharpoons \{\varphi \mid Prf(n, \ulcorner\varphi\urcorner)\}$ is finite and the function $n \mapsto Th(n)$ is total recursive;

3) for every finite set of arithmetical theorems Γ there exists a natural number n such that $\Gamma \subseteq Th(n)$.

Lemma 9. *Let L be a language from definition 1. For every pair of normal proof predicates Prf_1, Prf_2 and every operation of L there exist a total recursive function which satisfies the corresponding axiom. For example, there exists a function app_i such that for all natural numbers k, n for all arithmetical sentences φ, ψ*

$$\mathsf{PA} \vdash Prf_i(k, \ulcorner\varphi \to \psi\urcorner) \to (Prf_i(n, \ulcorner\varphi\urcorner) \to Prf_i(app_i(k,n), \ulcorner\psi\urcorner))$$

Definition 7. *Let L be one of the languages from definition 2. An arithmetical interpretation $* = (Prf_1, Prf_2, (\cdot)^*)$ for the language L has the following parameters:*

- *two normal proof predicate Prf_1 and Prf_2;*
- *total recursive functions for operations of L which satisfy lemma 9*
- *an evaluation $(\cdot)^*$ that assigns natural numbers to proof variables and arithmetical sentences to propositional variables.*

Arithmetical evaluation $(\cdot)^*$ can be extended to all LP^2 terms and formulas in the following way. It commutes with the Boolean connectives and

$$([\![t]\!]_i A)^* \rightleftharpoons \exists x\ (i = i \wedge x = \ulcorner A^* \urcorner \wedge Prf_i(t^*, x)).$$

Note that $\mathsf{PA} \vdash ([\![p]\!]A)^* \leftrightarrow Prf(p^*, \ulcorner A^* \urcorner)$. The reasons why we interpret the proof predicates in the more sophisticated way is that it makes the following problem decidable: being given Prf_i, an arithmetical formula φ and an $\mathcal{L}$-formula F, decide whether there exists $(\cdot)^*$, such that $F^* = \varphi$. If such $*$ exists then it is unique.

Theorem 5. *[Arithmetical soundness and completeness]*
For every LP^2 *formula A the following three propositions are equivalent:*

1) $\mathsf{LP}^2 \vdash A$*;*

2) for every interpretation $*$, $\mathsf{PA} \vdash A^*$*;*

3) for every interpretation $*$, A^* *is true.*

References

1. S. Artemov, *Uniform provability realization of intuitionistic logic, modality and λ-terms*, Electronic Notes in Theoretical Computer Science 23 (1999).
2. S. Artemov, *Explicit provability and constructive semantics*, Bulletin of Symbolic Logic 7 (2001), 1–36.
3. S. Artemov, *Evidence–based common knowledge*, Technical Report TR–2004018, CUNY Ph.D. Program in Computer Science, 2005.
4. M. Fitting, *The Logic of Proofs, Semantically*, Annals of Pure and Applied Logic, vol. 132 (2005), no. 1, pp. 1-25.
5. R. Kuznets, *On the complexity of explicit modal logics*, Computer Science Logic 2000, Lecture Notes in Computer Science 1862 (2000), 371–383.
6. A. Mkrtychev, *Models for the Logic of Proofs*, Lecture Notes in Computer Science, v. 1234 (1997), *Logical Foundations of Computer Science '97, Yaroslavl'*, pp. 266–275.
7. C. Smoryński, *Self-reference and Modal Logic*, Springer, New York, 1985.