Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 300 (2014) 89–103

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Intuitionistic Hypothetical Logic of Proofs

Gabriela Steren[1,2]

Computer Science Department
Universidad de Buenos Aires
Argentina

Eduardo Bonelli[3]

UNQ and CONICET
Argentina

Abstract

We study a term assignment for an intuitonistic fragment of the Logic of Proofs (LP). LP is a refinement of modal logic S4 in which the assertion $\Box A$ is replaced by $[\![s]\!]A$ whose intended reading is "*s is a proof of A*". We first introduce a natural deduction presentation based on hypothetical judgements and then its term assignment, which yields a confluent and strongly normalising typed lambda calculus λ^{IHLP}. This work is part of an ongoing effort towards reformulating LP in terms of hypothetical reasoning in order to explore its applications in programming languages.

Keywords: Curry-Howard, Logic of Proofs, Lambda Calculus, Programming Languages

1 Introduction

This paper is part of our ongoing exploration of the applications of Sergei Artemov's Logic of Proofs LP [2,4] in foundations of programming languages and type theory by means of the Curry-de Bruijn- Howard isomorphism. LP is a refinement of S4 in which $\Box A$ is replaced by $[\![s]\!]A$ and whose intended reading is "*s is a proof of A*". It has its roots in Provability Logic, and is one possible approach to the formalisation of the BHK interpretation of Intuitionistic Logic given that (1) it realizes all S4 theorems; and (2) is arithmetically sound and complete. One interesting feature of LP is that it is capable of reflecting its own derivations in the sense that if a formula A is provable, then $[\![s]\!]A$ is also provable, where s encodes a derivation of A. The

[1] Work partially supported by STIC-AmSud project 12STIC-04 – "Formal Development of Computer Programs and Applications" and Instituto Tecnológico de Buenos Aires (ITBA).

[2] Email: gsteren@yahoo.com

[3] Email: ebonelli@unq.edu.ar

http://dx.doi.org/10.1016/j.entcs.2013.12.013

aforementioned exploration aims at proposing natural deduction presentations of LP and their corresponding term assignment, with the hope of obtaining computational formalisms that cater both for terms and type derivations in a *unified* setting.

This work adds to previous results that we have developed [6,10,12]. Here we propose a natural deduction presentation of ILP, an intuitionistic fragment of LP, based on a judgemental analysis of modal logic [15,16,21] which includes the plus proof polynomial constructor of LP and also explore a variant of the term assignment of [6]. Judgements of ILP take the form $\Theta; \Gamma \vdash A \mid s$ (read "A is true with proof witness s under truth hypotheses Θ and validity hypotheses Γ"). Their meaning is given by appropriate axiom and inference schemes.

The paper is structured as follows. Sec. 2 introduces IHLP, a natural deduction presentation of ILP. We then study the correspondence with ILP in Sec. 3. Sec. 4 presents the term assignment, λ^{IHLP}, and then shows subject reduction, strong normalisation and confluence. Sec. 5 discusses related work. Finally, we conclude and suggest avenues for further research. For further details please consult [25].

2 IHLP

Formulae and proof witnesses of IHLP are given by the following grammar:

$$\begin{aligned} A, B &::= P \mid A \supset B \mid A \wedge B \mid A \vee B \mid [\![s]\!]A \\ r, s, t &::= x^A \mid v^A \mid \lambda x^A.s \mid s \cdot t \mid \langle s, t\rangle \mid \mathsf{fst}(s) \mid \mathsf{snd}(s) \\ &\quad \mid \ \mathsf{inl}(s) \mid \mathsf{inr}(s) \mid \mathsf{case}\, r\, [x^A].s\, [y^B].t \mid !s \mid \textsc{LetB}\, v^A\, \textsc{be}\, r, s\, \textsc{in}\, t \mid s + t \end{aligned}$$

where $P, Q, \ldots$ ranges over a set of propositional variables, $x^A, y^A, z^A, \ldots$ over a set of *truth* variables and $u^A, v^A, \ldots$ over a set of *validity* variables. A formula may either be a propositional variable, an implication $A \supset B$, a conjunction $A \wedge B$, a disjunction $A \vee B$ or a modality $[\![s]\!]A$. A proof witness may either be a truth or validity variable, an abstraction $\lambda x^A.s$, an application $s \cdot t$, a pair $\langle s, t\rangle$, projections $\mathsf{fst}(s)$ and $\mathsf{snd}(s)$, injections $\mathsf{inl}(s)$ and $\mathsf{inr}(s)$, a case $\mathsf{case}\, r\, [x^A].s\, [y^B].t$, a bang $!s$, an unbox $\textsc{LetB}\, v^A\, \textsc{be}\, r, s\, \textsc{in}\, t$ or a plus $s + t$. We write $A\{x^A {:=}\, r\}$ (resp. $A\{v^A {:=}\, r\}$) for the capture-avoiding substitution of truth (resp. validity) variables for proof witnesses in formulae; similarly for substitution of truth/validity variables in proof terms $s\{x^A {:=}\, t\}$ (resp. $s\{v^A {:=}\, t\}$).

Free variables of validity $\mathsf{FVV}(s)$ and truth $\mathsf{FVT}(s)$ over a proof witness s are as expected. Some sample defining clauses are illustrated below, where $\mathsf{FVT}(t, s)$ abbreviates $\mathsf{FVT}(t) \cup \mathsf{FVT}(s)$. These definitions extend in the obvious way to formulae.

$$\begin{array}{ccc ccc} \mathsf{FVT}(x^A) & \stackrel{\mathrm{def}}{=} & \{x^A\} & \mathsf{FVV}(x^A) & \stackrel{\mathrm{def}}{=} & \emptyset \\ \mathsf{FVT}(v^A) & \stackrel{\mathrm{def}}{=} & \emptyset & \mathsf{FVV}(v^A) & \stackrel{\mathrm{def}}{=} & \{v^A\} \\ \mathsf{FVT}(!s) & \stackrel{\mathrm{def}}{=} & \mathsf{FVT}(s) & \mathsf{FVV}(!s) & \stackrel{\mathrm{def}}{=} & \mathsf{FVV}(s) \\ \mathsf{FVT}(\textsc{LetB}\, v^A\, \textsc{be}\, u{,}s\, \textsc{in}\, t) & \stackrel{\mathrm{def}}{=} & \mathsf{FVT}(t, s) & \mathsf{FVV}(\textsc{LetB}\, v^A\, \textsc{be}\, u{,}s\, \textsc{in}\, t) & \stackrel{\mathrm{def}}{=} & (\mathsf{FVV}(t) \setminus \{v^A\}) \cup \mathsf{FVV}(s) \\ \mathsf{FVT}(\lambda x^A.s) & \stackrel{\mathrm{def}}{=} & \mathsf{FVT}(s) \setminus \{x^A\} & \mathsf{FVV}(\lambda x^A.s) & \stackrel{\mathrm{def}}{=} & \mathsf{FVV}(s) \end{array}$$

Judgements take the form $\Theta; \Gamma \vdash A \mid s$ with validity context $\Theta = {v_1}^{A_1}, \ldots, {v_m}^{A_m}$, truth context $\Gamma = {x_1}^{B_1}, \ldots, {x_n}^{B_n}$, A a formula, and s a proof witness. We write "$\cdot$" for empty contexts. In a judgement, in addition to the

$$\frac{}{\Theta;\Gamma,x^A\vdash A\mid x^A}\,\mathsf{Var}\qquad\qquad \frac{}{\Theta,v^A;\Gamma\vdash A\mid v^A}\,\mathsf{VarM}$$

$$\frac{\Theta;\Gamma,x^A\vdash B\mid s}{\Theta;\Gamma\vdash A\supset B\mid \lambda x^A.s}\,{\supset}\mathsf{I}\qquad\qquad \frac{\Theta;\Gamma\vdash A\supset B\mid s\quad \Theta;\Gamma\vdash A\mid t}{\Theta;\Gamma\vdash B\mid s\cdot t}\,{\supset}\mathsf{E}$$

$$\frac{\Theta;\Gamma\vdash A\mid s\quad \Theta;\Gamma\vdash B\mid t}{\Theta;\Gamma\vdash A\wedge B\mid\langle s,t\rangle}\,\wedge_{\mathsf{I}}\qquad \frac{\Theta;\Gamma\vdash A\wedge B\mid s}{\Theta;\Gamma\vdash A\mid \mathsf{fst}(s)}\,\wedge_{\mathsf{E1}}\qquad \frac{\Theta;\Gamma\vdash A\wedge B\mid s}{\Theta;\Gamma\vdash B\mid \mathsf{snd}(s)}\,\wedge_{\mathsf{E2}}$$

$$\frac{\Theta;\Gamma\vdash A\mid s}{\Theta;\Gamma\vdash A\vee B\mid \mathsf{inl}(s)}\,\vee_{\mathsf{I1}}\qquad\qquad \frac{\Theta;\Gamma\vdash B\mid s}{\Theta;\Gamma\vdash A\vee B\mid \mathsf{inr}(s)}\,\vee_{\mathsf{I2}}$$

$$\frac{\Theta;\Gamma\vdash A\vee B\mid r\quad \Theta;\Gamma,x^A\vdash C\mid s\quad \Theta;\Gamma,y^B\vdash C\mid t}{\Theta;\Gamma\vdash C\mid \mathsf{case}\,r\,[x^A].s\,[y^B].t}\,\vee_{\mathsf{E}}$$

$$\frac{\Theta;\cdot\vdash A\mid s\quad \Theta;\cdot\vdash s\equiv t:A}{\Theta;\Gamma\vdash [\![t]\!]A\mid !t}\,\Box\mathsf{I}\qquad \frac{\Theta;\Gamma\vdash [\![r]\!]A\mid s\quad \Theta,v^A;\Gamma\vdash C\mid t}{\Theta;\Gamma\vdash C\{v^A:=r\}\mid \textsc{LetB}\,v^A\,\textsc{be}\,r,s\,\textsc{in}\,t}\,\Box\mathsf{E}$$

$$\frac{\Theta;\Gamma\vdash A\mid s}{\Theta;\Gamma\vdash A\mid s+t}\,\mathsf{PlusL}\qquad\qquad \frac{\Theta;\Gamma\vdash A\mid t}{\Theta;\Gamma\vdash A\mid s+t}\,\mathsf{PlusR}$$

Fig. 1. Axiom and inference schemes of IHLP (1/2)

requirement that the $v_i^{A_i}$ and $x_i^{B_i}$ be distinct, we also require that they be fresh (i.e. that they do not occur in the $A_1,\ldots,A_m$ and $B_1,\ldots,B_n$).

A judgement is said to be *derivable* if it may be inferred using the axiom and inference schemes of Fig. 1. Note that if a derivation π of a judgement $\Theta;\Gamma\vdash A\mid s$ is obtained using these axioms and inference schemes, then s does not necessarily determine π (due to $\Box\mathsf{I}$, PlusL and PlusR). Most of these axioms and inference schemes are self-explanatory. For example, the axiom VarM states that if A is assumed valid, then we can conclude that A is true. The salient schemes are $\Box\mathsf{I}$ and those for plus. The former is a generalization of the following simpler one, which is a natural explicit counterpart of the standard introduction scheme for the $\Box$ modality in the judgemental setting [15,16,21].

$$\frac{\Theta;\cdot\vdash A\mid s}{\Theta;\Gamma\vdash [\![s]\!]A\mid !s}\,\Box\mathsf{I}_0$$

Although sound, $\Box\mathsf{I}_0$ is not satisfactory from the point of view of normalisation of derivations. For example, consider the derivation on the left in Fig. 2, where $\pi_{1,2}$ are derivations of $\Theta;x^A\vdash B\mid s$ and $\Theta;\cdot\vdash A\mid t$, resp. and π_3 is obtained from an appropriate substitution principle. A normalisation step would produce the one on the right. However, this derivation is not valid since the proof witness in the hypothesis of $\Box\mathsf{I}_0$ must be identical to the one in the argument of "!" in the judgement in the conclusion. Indeed this is not the case, since on one hand we have $s\{x^A:=t\}$, while on the other we have $(\lambda x^A.s)\cdot t$. The introduction scheme $\Box\mathsf{I}$ for the modality remedies this situation by obtaining the derivation:

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\pi_1\\ \Theta; x^A \vdash B \,|\, s\end{array}}{\Theta; \cdot \vdash A \supset B \,|\, \lambda x^A.s}\;\supset\mathsf{I} \qquad \begin{array}{c}\pi_2\\ \Theta; \cdot \vdash A \,|\, t\end{array}}{\Theta; \cdot \vdash B \,|\, \Theta; \cdot \vdash B \,|\, (\lambda x^A.s) \cdot t}\;\supset\mathsf{E}}{\Theta; \Gamma \vdash \llbracket(\lambda x^A.s) \cdot t\rrbracket B \,|\, !((\lambda x^A.s) \cdot t)}\;\Box\mathsf{I}_0 \quad \to \quad \dfrac{\begin{array}{c}\pi_3\\ \Theta; \cdot \vdash B \,|\, s\{x^A{:=}\, t\}\end{array}}{\Theta; \Gamma \vdash \llbracket(\lambda x^A.s) \cdot t\rrbracket B \,|\, !((\lambda x^A.s) \cdot t)}\;\Box\mathsf{I}_0$$

Fig. 2. Failure of SR in the presence of $\Box\mathsf{I}_0$

$$\dfrac{\Theta; \Gamma, x^A \vdash B \,|\, s \quad \Theta; \Gamma \vdash A \,|\, t}{\Theta; \Gamma \vdash (\lambda x^A.s) \cdot t \equiv s\{x^A := t\} : B}\;\mathsf{Eq}\text{-}\beta$$

$$\dfrac{\Theta; \cdot \vdash A \,|\, s \quad \Theta, v^A; \Gamma \vdash C \,|\, t}{\Theta; \Gamma \vdash \textsc{LetB}\, v^A \,\textsc{be}\, s, (!s) \,\textsc{in}\, t \equiv t\{v^A := s\} : C\{v^A := s\}}\;\mathsf{Eq}\text{-}\gamma$$

$$\dfrac{\Theta; \Gamma \vdash A \supset B \,|\, r \quad \Theta; \Gamma \vdash A \,|\, t}{\Theta; \Gamma \vdash (r + s) \cdot t \equiv (r \cdot t) + s : B}\;\mathsf{Eq}\text{-}\psi_L \qquad \dfrac{\Theta; \Gamma \vdash A \supset B \,|\, s \quad \Theta; \Gamma \vdash A \,|\, t}{\Theta; \Gamma \vdash (r + s) \cdot t \equiv r \cdot (s + t) : B}\;\mathsf{Eq}\text{-}\psi_R$$

$$\dfrac{\Theta; \Gamma \vdash \llbracket r\rrbracket A \,|\, s \quad \Theta, v^A; \Gamma \vdash C \,|\, q}{\Theta; \Gamma \vdash \textsc{LetB}\, v^A \,\textsc{be}\, r, (s + t) \,\textsc{in}\, q \equiv \textsc{LetB}\, v^A \,\textsc{be}\, r, s \,\textsc{in}\, q + t : C\{v^A := r\}}\;\mathsf{Eq}\text{-}\phi_L$$

$$\dfrac{\Theta; \Gamma \vdash \llbracket r\rrbracket A \,|\, t \quad \Theta, v^A; \Gamma \vdash C \,|\, q}{\Theta; \Gamma \vdash \textsc{LetB}\, v^A \,\textsc{be}\, r, (s + t) \,\textsc{in}\, q \equiv s + \textsc{LetB}\, v^A \,\textsc{be}\, r, t \,\textsc{in}\, q : C\{v^A := r\}}\;\mathsf{Eq}\text{-}\phi_R$$

Fig. 3. Axiom and inference schemes of IHLP (2/2)

$$\dfrac{\dfrac{\pi_3}{\Theta; \cdot \vdash B \,|\, s\{x^A{:=}\, t\}} \qquad \Theta; \cdot \vdash s\{x^A{:=}\, t\} \equiv (\lambda x^A.s) \cdot t : A}{\Theta; \Gamma \vdash \llbracket(\lambda x^A.s) \cdot t\rrbracket B \,|\, !((\lambda x^A.s) \cdot t)}\;\Box\mathsf{I}$$

A sample of the axiom and inference schemes defining the judgement $\Theta; \Gamma \vdash s \equiv t : A$ are depicted in Fig. 3 (see [25] for the full set). These schemes are closely tied to the normalisation relation on derivations. Indeed, since LP is capable of reflecting its own derivations and these derivations are equated by normalisation, the induced relation between derivations must also be formalised in the logic itself. It should be noted that LP was originally formulated in a Hilbert-style presentation, which does not allow such an observation to be made.

Regarding the schemes for plus, they are a consequence of the fact that LP is a *multi-conclusion* logic in the sense that a proof witness may prove more than one formula. Indeed, note that the following holds in LP: $\llbracket s\rrbracket A \wedge \llbracket t\rrbracket B \supset \llbracket s+t\rrbracket A \wedge \llbracket s+t\rrbracket B$ and hence $s+t$ proves both A and B. In the particular case that A and B coincide, $s + t$ denotes two proofs of A. This non-deterministic conjunction of proofs is necessary to be able to *realize* all theorems of IS4 (cf. Sec 5). By *realize* [4, Def.9.2] we mean decorate the boxes of IS4 theorems so that the resulting formulae are provable in ILP. As an example, we show how the IS4 theorem $\Box A \vee \Box B \supset \Box(A \vee B)$ may be realized in IHLP as $\llbracket s\rrbracket A \vee \llbracket t\rrbracket B \supset \llbracket \mathsf{inl}(s) + \mathsf{inr}(t)\rrbracket(A \vee B)$, for any s and t.

Example 2.1 Let $\Theta_1 \stackrel{\text{def}}{=} v^A$, $\Theta_2 \stackrel{\text{def}}{=} u^B$, $\Gamma \stackrel{\text{def}}{=} z^{[\![s]\!]A\vee[\![t]\!]B}$ $\Gamma_1 \stackrel{\text{def}}{=} z^{[\![s]\!]A\vee[\![t]\!]B}, x^{[\![s]\!]A}$ and $\Gamma_2 \stackrel{\text{def}}{=} z^{[\![s]\!]A\vee[\![t]\!]B}, y^{[\![t]\!]B}$ in the following two derivations $\pi_{1,2}$:

$$\dfrac{\cdot;\Gamma_1 \vdash [\![s]\!]A \mid x^{[\![s]\!]A} \qquad \dfrac{\dfrac{\dfrac{\Theta_1;\cdot \vdash A \mid v^A}{\Theta_1;\cdot \vdash A\vee B \mid \mathsf{inl}(v^A)}}{\Theta_1;\cdot \vdash A\vee B \mid \mathsf{inl}(v^A)+\mathsf{inr}(t)}\,\mathsf{PlusL}}{\Theta_1;\Gamma_1 \vdash [\![\mathsf{inl}(v^A)+\mathsf{inr}(t)]\!](A\vee B) \mid !(\mathsf{inl}(v^A)+\mathsf{inr}(t))}\,\Box\mathsf{I}}{\cdot;\Gamma_1 \vdash [\![\mathsf{inl}(s)+\mathsf{inr}(t)]\!](A\vee B) \mid \textsc{LetB}\, v^A \,\textsc{be}\, s, x^{[\![s]\!]A} \,\textsc{in}\, !(\mathsf{inl}(v^A)+\mathsf{inr}(t))}\,\Box\mathsf{E}$$

$$\dfrac{\cdot;\Gamma_2 \vdash [\![t]\!]B \mid y^{[\![s]\!]B} \qquad \dfrac{\dfrac{\dfrac{\Theta_2;\cdot \vdash B \mid u^B}{\Theta_2;\cdot \vdash A\vee B \mid \mathsf{inr}(u^B)}\,\vee_{\mathsf{I2}}}{\Theta_2;\cdot \vdash A\vee B \mid \mathsf{inl}(s)+\mathsf{inr}(u^B)}\,\mathsf{PlusR}}{\Theta_2;\Gamma_2 \vdash [\![\mathsf{inl}(s)+\mathsf{inr}(u^B)]\!](A\vee B) \mid !(\mathsf{inl}(s)+\mathsf{inr}(u^B))}\,\Box\mathsf{I}}{\cdot;\Gamma_2 \vdash [\![\mathsf{inl}(s)+\mathsf{inr}(t)]\!](A\vee B) \mid \textsc{LetB}\, v^A \,\textsc{be}\, t, y^{[\![s]\!]B} \,\textsc{in}\, !(\mathsf{inl}(s)+\mathsf{inr}(u^B))}\,\Box\mathsf{E}$$

Finally, for π_3 below consider the definitions
$r_1 \stackrel{\text{def}}{=} \textsc{LetB}\, v^A \,\textsc{be}\, s, x^{[\![s]\!]A} \,\textsc{in}\, !(\mathsf{inl}(v^A)+\mathsf{inr}(t))$,
$r_2 \stackrel{\text{def}}{=} \textsc{LetB}\, u^B \,\textsc{be}\, t, y^{[\![t]\!]B} \,\textsc{in}\, !(\mathsf{inl}(s)+\mathsf{inr}(u^B))$, and
$r_3 \stackrel{\text{def}}{=} \mathsf{case}\; z^{[\![s]\!]A\vee[\![t]\!]B}\,[x^A].r_1\;[y^B].r_2$.

$$\dfrac{\cdot;\Gamma\vdash[\![s]\!]A\vee[\![t]\!]B \mid z^{[\![s]\!]A\vee[\![t]\!]B} \quad \overset{\pi_1}{\cdot;\Gamma_1\vdash[\![\mathsf{inl}(s)+\mathsf{inr}(t)]\!](A\vee B)\mid r_1} \quad \overset{\pi_2}{\cdot;\Gamma_2\vdash[\![\mathsf{inl}(s)+\mathsf{inr}(t)]\!](A\vee B)\mid r_2}}{\cdot;\Gamma\vdash[\![\mathsf{inl}(s)+\mathsf{inr}(t)]\!](A\vee B)\mid r_3}\,\vee_{\mathsf{E}}$$

Note that the use of PlusL in π_1 and PlusR in π_2 is required in order to concatenate the two alternative proofs of $A\vee B$ into a unique non-deterministic proof, and allow the application of $\vee_{\mathsf{E}}$ in π_3.

Remark 2.2 One may wonder whether, for the implicative fragment, the plus may be dispensed with while still maintaining realization of all S4 theorems. This is the case if, in the terminology of LP, so called non-injective specification sets and non-normal realizations are allowed (see [18] and also [5, Sec.11.2]).

The following basic results are proved by induction on the derivation.

Lemma 2.3

- *(Weakening) If the judgement* $\Theta;\Gamma\vdash A\mid s$ *is derivable, then so is* $\Theta\cup\Theta';\Gamma\cup\Gamma'\vdash A\mid s$.
- *(Strengthening) If the judgement* $\Theta;\Gamma\vdash A\mid s$ *is derivable, then so is the judgement* $\Theta\cap \mathit{FVV}(s);\Gamma\cap \mathit{FVT}(s)\vdash A\mid s$.
- *(Substitution of Truth Variables) If* $\Theta;\Gamma,x^A\vdash B\mid s$ *and* $\Theta;\Gamma\vdash A\mid t$ *are derivable, then so is* $\Theta;\Gamma\vdash B\mid s\{x^A:=t\}$.
- *(Substitution of Validity Variables) If* $\Theta,v^A;\Gamma\vdash B\mid s$ *and* $\Theta;\cdot\vdash A\mid t$ *are derivable, then so is* $\Theta;\Gamma\vdash B\{v^A:=t\}\mid s\{v^A:=t\}$.

3 Relating ILP and IHLP

This section addresses the relation between ILP and IHLP. We begin by recalling the definition of ILP and then state the required results, restricting our attention to the implicative fragment. Then we show that all ILP theorems are derivable in IHLP (Prop. 3.1) and conversely that all judgements derivable in IHLP may be translated to judgements derivable in ILP (Prop. 3.9).

Assume given a set of *proof constants* $\mathcal{C}$ and $c \in \mathcal{C}$. The formulae of ILP are those of IHLP except that the proof witnesses encode Hilbert-style proofs and are called *proof polynomials* [2,4]:

$$s, t ::= x^A \mid c \mid s \cdot t \mid !s \mid s + t$$

The axioms and inference schemes of ILP are as follows, where a context Γ is a set of hypotheses of the form x^A and we assume[4] that $\mathcal{C}$ includes at least one constant for each instance of an axiom scheme **A0-A5**:

A0. Axioms of minimal propositional logic in the language of ILP.

A1. $\llbracket t \rrbracket A \supset A$

A2. $\llbracket s \rrbracket (A \supset B) \supset (\llbracket t \rrbracket A \supset \llbracket s \cdot t \rrbracket B)$

A3. $\llbracket t \rrbracket A \supset \llbracket !t \rrbracket \llbracket t \rrbracket A$

A4. $\llbracket s \rrbracket A \supset \llbracket s + t \rrbracket A$

A5. $\llbracket t \rrbracket A \supset \llbracket s + t \rrbracket A$

R1. $\Gamma \vdash A \supset B$ and $\Gamma \vdash A$ implies $\Gamma \vdash B$. (*MP*)

R2. If A is an axiom **A0-A5**, and $c \in \mathcal{C}$ corresponds to A, then $\Gamma \vdash \llbracket c \rrbracket A$. (*Necessitation*)

The translation $\underline{\bullet}$ from ILP formulae and proof polynomials to IHLP formulae and proof witnesses is simply the structure preserving mapping that replaces all occurrences of proof constants by IHLP proof witnesses that prove the corresponding axioms (cf. [25]). Some sample defining clauses are:

$$\begin{aligned}
\underline{c^{\mathbf{A0}}_{A,B}} &\stackrel{\text{def}}{=} (\lambda x^{\underline{A}}.\lambda y^{\underline{B}}.x) \\
\underline{c^{\mathbf{A1}}_{t,A}} &\stackrel{\text{def}}{=} \lambda x^{\underline{\llbracket t \rrbracket A}}.\textsc{LetB}\, v^{\underline{A}}\, \textsc{be}\, \underline{t}, x\, \textsc{in}\, v \\
\underline{c^{\mathbf{A2}}_{s,t,A,B}} &\stackrel{\text{def}}{=} \lambda x^{\underline{\llbracket s \rrbracket A \supset B}}.\lambda y^{\underline{\llbracket t \rrbracket A}}.\textsc{LetB}\, v^{\underline{A}}\, \textsc{be}\, \underline{t}, y\, \textsc{in}\, \textsc{LetB}\, w^{\underline{A \supset B}}\, \textsc{be}\, \underline{s}, x\, \textsc{in}\, !(w \cdot v) \\
\underline{c^{\mathbf{A3}}_{s,A}} &\stackrel{\text{def}}{=} \lambda x^{\underline{\llbracket s \rrbracket A}}.\textsc{LetB}\, v^{\underline{A}}\, \textsc{be}\, \underline{s}, x^{\underline{\llbracket s \rrbracket A}}\, \textsc{in}\, !!v^{\underline{A}} \\
\underline{c^{\mathbf{A4}}_{s,t,A}} &\stackrel{\text{def}}{=} \lambda x^{\underline{\llbracket s \rrbracket A}}.\textsc{LetB}\, v^{\underline{A}}\, \textsc{be}\, \underline{s}, x\, \textsc{in}\, !(\underline{v} + \underline{t})
\end{aligned}$$

It extends naturally to contexts of hypotheses $\underline{\Gamma} \stackrel{\text{def}}{=} \{x^{\underline{A}} \text{ s.t. } x^A \in \Gamma\}$.

Proposition 3.1 *If $\Gamma \vdash F$ is derivable in ILP, then so is $\cdot; \underline{\Gamma} \vdash \underline{F} \mid s$ in IHLP for some proof witness s.*

[4] More general assumptions are possible. See [2,4].

Remark 3.2 In PlusL, no requirements on the truth or validity variables of t are assumed in relation to the contexts Θ and Γ. An alternative inference scheme for PlusL (and similarly for PlusR) might be:

$$\frac{\Theta; \Gamma \vdash A \,|\, s \quad \mathsf{FVV}(t) \subseteq \Theta \quad \mathsf{FVT}(s) \subseteq \Gamma}{\Theta; \Gamma \vdash A \,|\, s + t}\ \mathsf{PlusL}$$

However, this scheme does not allow the proof of proposition 3.1 to go through in the case of axiom **A4**, since no restriction is a priori placed on t in that axiom, and $\Box\mathsf{I}$ requires that there be no truth dependencies. It may be possible to retain the alternative scheme proposed above, by drawing ideas from Contextual Modal Type Theory [19].

Suppose Γ is the context $\{x_1^{A_1}, \ldots, x_n^{A_n}\}$ and $\boldsymbol{s} = s_1, \ldots . s_n$. Then we write $[\![\boldsymbol{s}]\!]\Gamma$ for the context $\{x_1^{[\![s_1]\!]A_1}, \ldots, x_n^{[\![s_n]\!]A_n}\}$.

Definition 3.3 Let π be a derivation in ILP of $[\![\boldsymbol{s}]\!]\Gamma \vdash F$. The *extracted witness of* π, denoted r below, is defined by induction on the length n of π. Suppose that $n = 1$. Then either F is an instance of an axiom or is a hypothesis in Γ. In the former we analyse each case:

- $F = A \supset B \supset A$ or $F = (A \supset B \supset C) \supset (A \supset B) \supset A \supset C$, then $r \overset{\text{def}}{=} c^{\mathbf{A0}}_{A,B}$ or $r \overset{\text{def}}{=} c^{\mathbf{A0}}_{A,B,C}$, resp.
- $F = [\![t]\!]A \supset A$, then $r \overset{\text{def}}{=} c^{\mathbf{A1}}_{t,A}$.
- $F = [\![s]\!](A \supset B) \supset ([\![t]\!]A \supset [\![s \cdot t]\!]B)$, then $r \overset{\text{def}}{=} c^{\mathbf{A2}}_{s,t,A,B}$.
- $F = [\![t]\!]A \supset [\![!t]\!][\![t]\!]A$, then $r \overset{\text{def}}{=} c^{\mathbf{A3}}_{t,A}$.
- $F = [\![s]\!]A \supset [\![s + t]\!]A$, then $r \overset{\text{def}}{=} c^{\mathbf{A4}}_{s,t,A}$.
- $F = [\![t]\!]A \supset [\![s + t]\!]A$, then $r \overset{\text{def}}{=} c^{\mathbf{A5}}_{s,t,A}$.

In the latter case (i.e. F is a hypothesis, say $[\![s]\!]B$ in $[\![\boldsymbol{s}]\!]\Gamma$), we set $r \overset{\text{def}}{=} !s$. For the inductive case, we consider each possible case for the last step:

- It is an axiom or a hypothesis, then we proceed as above.
- F is obtained from formulae $F_{1,2}$ using MP. Let $r_{1,2}$ be the witnesses extracted from the derivations ending in $F_{1,2}$. Set $r \overset{\text{def}}{=} r_1 \cdot r_2$.
- F is obtained from an application of Necessitation: $F = [\![c]\!]F_1$, where F_1 is an instance of an axiom $\mathbf{A}$. Then we set $r \overset{\text{def}}{=} !c^{\mathbf{A}}$.

The following result, proved by induction on $[\![\boldsymbol{s}]\!]\Gamma \vdash A$, states that ILP can internalise its own derivations.

Lemma 3.4 (Internalisation) *Suppose $[\![\boldsymbol{s}]\!]\Gamma \vdash A$ is a derivable judgement in ILP with a derivation π. Then $[\![\boldsymbol{s}]\!]\Gamma \vdash [\![r]\!]A$ is derivable, where r is the proof witness extracted from π.*

Let us write $\pi(\Gamma \vdash A)$ to denote that π is an ILP-derivation of $\Gamma \vdash A$. The following result shows how the deduction lemma can be internalized. Its proof relies on a lemma called the Stripping Lemma, which is stated below.

Lemma 3.5 (λ-Abstraction) *If $\llbracket \boldsymbol{u} \rrbracket \Gamma, y^{\llbracket x^A \rrbracket A} \vdash \llbracket s(\boldsymbol{u}, x^A) \rrbracket B$ is derivable and $x^A \notin \Gamma, B$, then there exists $t_\lambda^{A \supset B}(\llbracket \boldsymbol{u} \rrbracket \Gamma)$ such that $\llbracket \boldsymbol{u} \rrbracket \Gamma \vdash \llbracket t_\lambda^{A \supset B}(\llbracket \boldsymbol{u} \rrbracket \Gamma) \rrbracket (A \supset B)$, and $t_\lambda^{A \supset B}(\llbracket \boldsymbol{u} \rrbracket \Gamma)$ is an extracted witness of $A \supset B$ in $\llbracket \boldsymbol{u} \rrbracket \Gamma$.*

Lemma 3.6 (Stripping) *Suppose π is an ILP-derivation of $\Gamma, x^{\llbracket y^A \rrbracket A} \vdash B$ and $y^A \notin \Gamma$. Then there is a derivation of $\Gamma, y^A \vdash B'$, where B' results from B, by replacing all occurrences of $\llbracket t \rrbracket A$ by A for every proof term t containing y^A (including constants for instances of axioms containing y^A).*

One last result shall be required for the proof of our main result (Prop. 3.9).

Lemma 3.7 (Substitution) *$\Gamma \vdash \llbracket s \rrbracket A$ and $\Gamma, y^{\llbracket x^A \rrbracket A} \vdash B$ and $x^A \notin \Gamma$ implies $\Gamma \vdash B\{x^A := s\}$.*

A proof witness s is *provable* if for some Θ, Γ and A, the judgement $\Theta; \Gamma \vdash A \mid s$ is derivable. The translation from IHLP formulae and proof witnesses to ILP formulae and proof polynomials is as follows, where $c^{\mathbf{A1}}$ is the proof constant denoting any instance of **A1**:

$$P^\star \stackrel{\text{def}}{=} P$$

$$(A \supset B)^\star \stackrel{\text{def}}{=} A^\star \supset B^\star$$

$$(\llbracket s \rrbracket A)^\star \stackrel{\text{def}}{=} \begin{cases} \llbracket s^\star \rrbracket A^\star, & \text{if } s \text{ is provable} \\ \llbracket c^{\mathbf{A1}} \cdot c^{\mathbf{A1}} \rrbracket A^\star, & \text{if } s \text{ is not provable} \end{cases}$$

$$x^{A^\star} \stackrel{\text{def}}{=} x^{A^\star}$$

$$v^{A^\star} \stackrel{\text{def}}{=} v^{A^\star}$$

$$(s+t)^\star \stackrel{\text{def}}{=} s^\star + t^\star$$

$$(s \cdot t)^\star \stackrel{\text{def}}{=} s^\star \cdot t^\star$$

$$(!s)^\star \stackrel{\text{def}}{=} !(s^\star)$$

$$(\textsc{LetB}\, v^A \,\textsc{be}\, r, s \,\textsc{in}\, t)^\star \stackrel{\text{def}}{=} t^\star\{v^{A^\star} := r^\star\}$$

$$(\lambda x^A.s)^\star \stackrel{\text{def}}{=} t_\lambda^{A^\star \supset B^\star}(\Theta^\star \cup \Gamma^\star) \text{ if } \exists \Theta, \Gamma \text{ s.t. } \Theta; \Gamma \vdash A \supset B \mid \lambda x^A.s \text{ derivable}$$

$$\cdot^\star \stackrel{\text{def}}{=} \cdot$$

$$(\Theta, v^A)^\star \stackrel{\text{def}}{=} \Theta^\star, \llbracket v^{A^\star} \rrbracket A^\star$$

$$(\Gamma, x^A)^\star \stackrel{\text{def}}{=} \Gamma^\star, \llbracket x^{A^\star} \rrbracket A^\star$$

$$(\Theta; \Gamma \vdash A \mid s)^\star \stackrel{\text{def}}{=} \Theta^\star \cup \Gamma^\star \vdash \llbracket s^\star \rrbracket A^\star$$

Remark 3.8 Sometimes there is more than one possible translation for a proof witness or a formula (for instance, $\lambda x^A.(y^B + z^B)$ and $\llbracket \lambda x^A.(y^B + z^B) \rrbracket A \supset B$). This happens only when the proof witness in question, or some witness within the formula, contains abstractions. By Internalization, if $\Theta^\star \cup \Gamma^\star \vdash A^\star \supset B^\star$ is derivable, then *any* extracted witness t of $A^\star \supset B^\star$ in $\Theta^\star \cup \Gamma^\star$ will suffice to derive $\Theta^\star \cup \Gamma^\star \vdash \llbracket t \rrbracket A^\star \supset B^\star$. This means that t can be chosen freely among all possible extracted witnesses. So, whenever a formula or proof witness appears more than once within a derivation and multiple translations exist for it, it is always possible to use the same translation in all cases by choosing the extracted witnesses in the same way. If different contexts have been used to obtain the formula/proof witness in each case, we can use Weakening to make the contexts coincide (by taking the union of the contexts used in all cases).

Proposition 3.9 *For every derivable judgement $\Theta; \Gamma \vdash A \,|\, s$ in* IHLP*, the* ILP*-judgement $\Theta^\star \cup \Gamma^\star \vdash \llbracket s^\star \rrbracket A^\star$ is derivable in* ILP*.*

Corollary 3.10 *If $\cdot; \cdot \vdash A \,|\, s$ is derivable in* IHLP*, then both $\cdot \vdash \llbracket s^\star \rrbracket A^\star$ and $\cdot \vdash A^\star$ are derivable in* ILP*.*

4 λ^{IHLP} – Syntax and Semantics

We study a term assignment for IHLP, dubbed λ^{IHLP}, together with the reduction rules over the set of terms which mimic normalisation of derivations in IHLP and address subject reduction, strong normalisation (SN) and confluence.

The set of terms for IHLP is defined as follows:

$$\begin{array}{rl} M, N ::= & x^A \,|\, v^A \,|\, (\lambda x^A.M^B)^{A \supset B} \,|\, (M^{A \supset B} N^A)^B \,|\, (\langle M, N \rangle)^{A \wedge B} \,|\, \mathsf{fst}(M^{A \wedge B})^A \\ | & \mathsf{snd}(M^{A \wedge B})^B \,|\, \mathsf{inl}(M)^{A \vee B} \,|\, \mathsf{inr}(M)^{A \vee B} \,|\, (\mathsf{case}\, M\, [x^A].P\, [y^B].Q)^C \\ | & (!M^A)^{\llbracket s \rrbracket A} \,|\, (\textsc{LetB}\, v^A \,\textsc{be}\, r, M^A \,\textsc{in}\, N^B)^{B\{v^A := r\}} \\ | & (M^A +_L s)^A \,|\, (s +_R N^B)^B \end{array}$$

Free variables of validity and truth for terms are defined analogously to those for proof witnesses. Type decorations are often omitted where it is safe. To the already introduced notions of substitution we add substitution of truth/validity variables in terms by proof witnesses/terms: $M^B\{a^A := N^A\}$ and $M^B\{v^A := N^A\}$. A *typing judgement* has the form $\Theta; \Gamma \vdash M^A \,|\, s$. The typing rules in Fig. 4 (obtained by assigning terms to the axiom and inference schemes of IHLP) define when a typing judgement is derivable. The following example term of type $\llbracket s \rrbracket A \vee \llbracket t \rrbracket B \supset \llbracket \mathsf{inl}(s) + \mathsf{inr}(t) \rrbracket (A \vee B)$ illustrates the term assigned to the derivation of Exm. 2.1:

$$\begin{array}{ll} \lambda z^{\llbracket s \rrbracket A \vee \llbracket t \rrbracket B}.\mathsf{case}\ z^{\llbracket s \rrbracket A \vee \llbracket t \rrbracket B} & [x^{\llbracket s \rrbracket A}].\textsc{LetB}\, v^A \,\textsc{be}\, s, x^{\llbracket s \rrbracket A} \,\textsc{in}\, !(\mathsf{inl}(v^A) + \mathsf{inr}(t)) \\ & [y^{\llbracket t \rrbracket B}].\textsc{LetB}\, u^B \,\textsc{be}\, t, y^{\llbracket t \rrbracket B} \,\textsc{in}\, !(\mathsf{inl}(s) + \mathsf{inr}(u^B)) \end{array}$$

Remark 4.1 Also in λ^{IHLP} (as already mentioned for IHLP), terms do not determine complete derivations due to the $\Box$I typing rule. For variations where this property does hold, see the discussion in Sec. 5.

λ^{IHLP}-reduction is defined as the compatible closure of the following two groups of *reduction rules*. The first set of rules, the *principal rules*, arises from the principal cases of normalisation of derivations.

$$\frac{}{\Theta;\Gamma,x^A \vdash x^A \mid x^A}\ \textsf{T-Var} \qquad \frac{}{\Theta,v^A;\Gamma \vdash v^A \mid v^A}\ \textsf{T-VarM}$$

$$\frac{\Theta;\Gamma,x^A \vdash M^B \mid s}{\Theta;\Gamma \vdash (\lambda x^A.M)^{A\supset B} \mid \lambda x^A.s}\ \textsf{T-}\supset\textsf{I} \qquad \frac{\Theta;\Gamma \vdash M^{A\supset B} \mid s \quad \Theta;\Gamma \vdash N^A \mid t}{\Theta;\Gamma \vdash (MN)^B \mid s\cdot t}\ \textsf{T-}\supset\textsf{E}$$

$$\frac{\Theta;\Gamma \vdash M^A \mid s \quad \Theta;\Gamma \vdash N^B \mid t}{\Theta;\Gamma \vdash \langle M,N\rangle^{A\wedge B} \mid \langle s,t\rangle}\ \textsf{T-}\wedge_{\textsf{I}} \qquad \frac{\Theta;\Gamma \vdash M^{A\wedge B} \mid s}{\Theta;\Gamma \vdash \textsf{fst}(M)^A \mid \textsf{fst}(s)}\ \textsf{T-}\wedge_{\textsf{E1}} \qquad \frac{\Theta;\Gamma \vdash M^{A\wedge B} \mid s}{\Theta;\Gamma \vdash \textsf{snd}(M)^B \mid \textsf{snd}(s)}\ \textsf{T-}\wedge_{\textsf{E2}}$$

$$\frac{\Theta;\Gamma \vdash M^A \mid s}{\Theta;\Gamma \vdash \textsf{inl}(M)^{A\vee B} \mid \textsf{inl}(s)}\ \textsf{T-}\vee_{\textsf{I1}} \qquad \frac{\Theta;\Gamma \vdash M^B \mid s}{\Theta;\Gamma \vdash \textsf{inr}(M)^{A\vee B} \mid \textsf{inr}(s)}\ \textsf{T-}\vee_{\textsf{I2}}$$

$$\frac{\Theta;\Gamma \vdash M^{A\vee B} \mid r \quad \Theta;\Gamma,x^A \vdash P^C \mid s \quad \Theta;\Gamma,y^B \vdash Q^C \mid t}{\Theta;\Gamma \vdash (\textsf{case}\,M\,[x^A].P\,[y^B].Q)^C \mid \textsf{case}\,r\,[x^A].s\,[y^B].t}\ \textsf{T-}\vee_{\textsf{E}}$$

$$\frac{\Theta;\cdot \vdash M^A \mid s \quad \Theta;\cdot \vdash s\equiv t : A}{\Theta;\Gamma \vdash (!M)^{\llbracket t\rrbracket A} \mid !t}\ \textsf{T-}\Box\textsf{I} \qquad \frac{\Theta;\Gamma \vdash M^{\llbracket r\rrbracket A} \mid s \quad \Theta,v^A;\Gamma \vdash N^C \mid t}{\Theta;\Gamma \vdash (\textsc{LetB}\,v^A\,\textsc{be}\,r,M\,\textsc{in}\,N)^{C\{v^A:=r\}} \mid \textsc{LetB}\,v^A\,\textsc{be}\,r,s\,\textsc{in}\,t}\ \textsf{T-}\Box\textsf{E}$$

$$\frac{\Theta;\Gamma \vdash M^A \mid s}{\Theta;\Gamma \vdash (M+_L t)^A \mid s+t}\ \textsf{T-PlusL} \qquad \frac{\Theta;\Gamma \vdash N^B \mid t}{\Theta;\Gamma \vdash (s+_R N)^B \mid s+t}\ \textsf{T-PlusR}$$

Fig. 4. Typing schemes

$$\begin{array}{lll}
\beta: & (\lambda x^A.M^B)N^A & \to M^B\{x^A := N^A\} \\
\beta_\Box: & \textsc{LetB}\,v^A\,\textsc{be}\,r,(!N^A)^{\llbracket t\rrbracket A}\,\textsc{in}\,M^B & \to M^B\{v^A := N^A\} \\
\rho_1: & \textsf{fst}(\langle M^A,N^B\rangle) & \to M^A \\
\rho_2: & \textsf{snd}(\langle M^A,N^B\rangle) & \to N^B \\
\delta_L: & \textsf{case}\,\textsf{inl}(M^A)^{A\vee B}\,[x^A].P^C\,[y^B].Q^C & \to P^C\{x^A := M^A\} \\
\delta_R: & \textsf{case}\,\textsf{inr}(M^B)^{A\vee B}\,[x^A].P^C\,[y^B].Q^C & \to Q^C\{y^B := M^B\}
\end{array}$$

The second set of rules, the *permutative rules*, arise from the permutative cases of normalisation. They simply permute all term constructs that encode an instance of an elimination scheme with the plus.

$$\begin{array}{ll}
\psi_L\colon (M^{A\supset B}+_L t)^{A\supset B}N^A & \to (M^{A\supset B}N^A)^B+_L t \\
\psi_R\colon (s+_R M^{A\supset B})^{A\supset B}N^A & \to s+_R(M^{A\supset B}N^A)^B \\
\phi_L\colon \textsc{LetB}\,v^A\,\textsc{be}\,r,(M^{\llbracket r\rrbracket A}+_L t)\,\textsc{in}\,N^B & \to (\textsc{LetB}\,v^A\,\textsc{be}\,r,M\,\textsc{in}\,N^B)^{B\{v^A:=r^A\}}+_L t \\
\phi_R\colon \textsc{LetB}\,v^A\,\textsc{be}\,r,(s+_R M^{\llbracket r\rrbracket A})\,\textsc{in}\,N^B & \to s+_R(\textsc{LetB}\,v^A\,\textsc{be}\,r,M\,\textsc{in}\,N^B)^{B\{v^A:=r^A\}} \\
\pi_L: \textsf{fst}((M^{A\wedge B}+_L s)^{A\wedge B})^A & \to (\textsf{fst}(M^{A\wedge B})^A+_L s)^A \\
\pi_R: \textsf{fst}((s+_R M^{A\wedge B})^{A\wedge B})^A & \to (s+_R\textsf{fst}(M^{A\wedge B})^A)^A \\
\sigma_L: \textsf{snd}((M^{A\wedge B}+_L s)^{A\wedge B})^B & \to (\textsf{snd}(M^{A\wedge B})^B+_L s)^B \\
\sigma_R: \textsf{snd}((s+_R M^{A\wedge B})^{A\wedge B})^B & \to (s+_R\textsf{snd}(M^{A\wedge B})^B)^B \\
\kappa_L\colon \textsf{case}\,(M^{A\vee B}+_L s)^{A\vee B}\,[x^A].P^C\,[y^B].Q^C & \to (\textsf{case}\,M^{A\vee B}\,[x^A].P^C\,[y^B].Q^C)^C+_L s \\
\kappa_R\colon \textsf{case}\,(s+_R M^{A\vee B})^{A\vee B}\,[x^A].P^C\,[y^B].Q^C & \to s+_R(\textsf{case}\,M^{A\vee B}\,[x^A].P^C\,[y^B].Q^C)^C
\end{array}$$

The first property we address is subject reduction.

Lemma 4.2 (Subject Reduction) *If $\Theta;\Gamma \vdash M^B \,|\, s$ is derivable and $M^B \to N^{B'}$, then $B' = B$ and $\Theta;\Gamma \vdash N^B \,|\, s'$ is derivable for some witness s' such that $\Theta;\Gamma \vdash s \equiv s' : B$.*

It would be tempting to expect that, $\Theta;\Gamma \vdash M^B \,|\, s$ is derivable and $M^B \to N^B$, then $\Theta;\Gamma \vdash N^B \,|\, s$ should also be derivable. However, this is not the case. For instance, $\cdot;\cdot \vdash ((\lambda x^{A\supset A}.x)\lambda y^A.y)^{A\supset A} \,|\, (\lambda x^{A\supset A}.x) \cdot \lambda y^A.y$ is derivable, but $\cdot;\cdot \vdash (\lambda y^A.y)^{A\supset A} \,|\, (\lambda x^{A\supset A}.x) \cdot \lambda y^A.y$ is not.

However, the above result holds for terms having a ! as their outermost operator.

Corollary 4.3 *If $\Theta;\Gamma \vdash (!M^B)^A | t$ is derivable and $M^B \to N^B$, then $\Theta;\Gamma \vdash (!N^B)^A \,|\, t$ is derivable.*

The above result gains significance in a programming setting where proof witnesses are used as certificates (see for example [12]), and all code must be certified in order to be executed. In this case, programs can be closed by an outer !, and thus full subject reduction is achieved.

Regarding strong normalisation, we define a mapping from λ^{IHLP}-terms into terms of the simply typed lambda calculus $\lambda^{\mathbf{1},\times,+}$ ($\mathbf{1}$ denotes the unit type) that preserves certain reduction properties. The result then follows from the fact that $\lambda^{\mathbf{1},\times,+}$ is strongly normalising [22].

The mapping $\llparenthesis \cdot \rrparenthesis$, associates types (formulae) and terms (proofs) in λ^{IHLP} with types and terms in $\lambda^{\mathbf{1},\times,+}$. It preserves the structure of formulae except in the case of the modal type $[\![s]\!]A$ which is mapped to a functional type whose domain is the unit type $\mathbf{1}$ and whose co-domain is the mapping of A (i.e. $\mathbf{1} \supset \llparenthesis A \rrparenthesis$). Both truth and validity variables are translated to the term variables of $\lambda^{\mathbf{1},\times,+}$. See [25] for full details.

Lemma 4.4 ($\llparenthesis \bullet \rrparenthesis$ preserves typability) *If $\Theta;\Gamma \vdash M^A \,|\, s$ is derivable in IHLP, then $\llparenthesis M \rrparenthesis : \llparenthesis \Theta \rrparenthesis \cup \llparenthesis \Gamma \rrparenthesis \vdash \llparenthesis A \rrparenthesis$ is derivable in $\lambda^{\mathbf{1},\times,+}$.*

Lemma 4.5 ($\llparenthesis \bullet \rrparenthesis$ commutes with substitution of truth variables)
For all λ^{IHLP}-terms M, N, for every truth variable x^A:
$\llparenthesis M \rrparenthesis \{x^{\llparenthesis A \rrparenthesis} := \llparenthesis N \rrparenthesis\} = \llparenthesis M\{x^A := N\} \rrparenthesis$.

Although $\llparenthesis \bullet \rrparenthesis$ does not commute with substitution of validity variables, the following result suffices for our purposes.

Lemma 4.6 *For all λ^{IHLP}-terms M, N, for every validity variable v^A and every truth variable $y^{\mathbf{1}} \notin \mathsf{FVT}(\llparenthesis N \rrparenthesis)$: $\llparenthesis M \rrparenthesis \{x_v^{\mathbf{1}\supset\llparenthesis A \rrparenthesis} := \lambda y^{\mathbf{1}}.\llparenthesis N \rrparenthesis\} \twoheadrightarrow_\beta \llparenthesis M\{v^A := N\} \rrparenthesis$.*

Lemma 4.7 *If $M \to N$ in IHLP* without *the use of permutative rules, then $\llparenthesis M \rrparenthesis \to^+ \llparenthesis N \rrparenthesis$ in $\lambda^{1,\times,+}$.*

Lemma 4.8 *If $M \to N$ in IHLP using* only *permutative rules, then $\llparenthesis M \rrparenthesis = \llparenthesis N \rrparenthesis$.*

By means of the following polynomial interpretation $(\cdot)_{\mathcal{A}}$ in $\mathbb{N}_{\geq 2}$, using the standard order for natural numbers, we can show SN of permutative reduction:

$$
\begin{aligned}
x^B_{\mathcal{A}} = v^B_{\mathcal{A}} &\stackrel{\text{def}}{=} 2\\
(M^{C\supset B} N^A)^B_{\mathcal{A}} &\stackrel{\text{def}}{=} M^{C\supset B}_{\mathcal{A}} \times N^A_{\mathcal{A}}\\
(\lambda x^A.M^B)^{A\supset B}_{\mathcal{A}} &\stackrel{\text{def}}{=} 2 \times M^B_{\mathcal{A}}\\
\langle M, N\rangle)^{A\wedge B}_{\mathcal{A}} &\stackrel{\text{def}}{=} M_{\mathcal{A}} + N_{\mathcal{A}}\\
\mathsf{fst}(M)^A_{\mathcal{A}} = \mathsf{snd}(M)^A_{\mathcal{A}} &\stackrel{\text{def}}{=} 2 \times M_{\mathcal{A}}\\
\mathsf{inl}(M)^{A\vee B}_{\mathcal{A}} = \mathsf{inr}(M)^{A\vee B}_{\mathcal{A}} &\stackrel{\text{def}}{=} M_{\mathcal{A}}\\
(\mathsf{case}\, M\, [x^A].P\, [y^B].Q)^C_{\mathcal{A}} &\stackrel{\text{def}}{=} 2 \times M_{\mathcal{A}} + P_{\mathcal{A}} + Q_{\mathcal{A}}\\
(!M^B)^{[\![s]\!]B}_{\mathcal{A}} &\stackrel{\text{def}}{=} 1 + M^B_{\mathcal{A}}\\
(\textsc{LetB}\, v^C\, \textsc{be}\, r, M\, \textsc{in}\, N^B)^{B\{v^C:=r\}}_{\mathcal{A}} &\stackrel{\text{def}}{=} N^B_{\mathcal{A}} \times M^{[\![r]\!]B}_{\mathcal{A}} + 1\\
(M^B +_L t)^B_{\mathcal{A}} &\stackrel{\text{def}}{=} 2 \times M^B_{\mathcal{A}} + 2\\
(s +_R M^B)^B_{\mathcal{A}} &\stackrel{\text{def}}{=} 2 \times M^B_{\mathcal{A}} + 2
\end{aligned}
$$

Lemma 4.9 *Permutative reduction is SN.*

We can now obtain SN for λ^{IHLP}.

Proposition 4.10 *Every typable* IHLP*-term is SN.*

Proof. By contradiction. Assume that there is an infinite reduction sequence starting from a typable λ^{IHLP}-term M_0. We will distinguish between principal reductions ($\xrightarrow{\text{B}}$) and permutative reductions ($\xrightarrow{\text{P}}$) within this sequence.

Since, by Lemma 4.9, permutative reduction is SN, our sequence must contain an infinite number of principal reduction steps. Between any two principal steps, there may be 0 or more permutative steps (always a finite number). Therefore, the reduction sequence has the form: $M_0 \overset{\text{P}}{\twoheadrightarrow} M'_0 \xrightarrow{\text{B}} M_1 \overset{\text{P}}{\twoheadrightarrow} M'_1 \xrightarrow{\text{B}} M_2 \overset{\text{P}}{\twoheadrightarrow} M'_2 \xrightarrow{\text{B}} \cdots$ Additionally, by Lemma 4.8, $(\!| M_i |\!) = (\!| M'_i |\!)$ for every i. Also, by Lemma 4.7, we know that for every i $(\!| M_i |\!) \to^+ (\!| M_{i+1} |\!)$ in $\lambda^{\mathbf{1},\times,+}$. We can therefore construct an infinite $\lambda^{\mathbf{1},\times,+}$-reduction sequence: $(\!| M_0 |\!) \to^+ (\!| M_1 |\!) \to^+ (\!| M_2 |\!) \to^+ \cdots$.

However, M_0 is typable in λ^{IHLP} and, by Lemma 4.2, so is every M_i. Since the mapping preserves typability (Lemma 4.4), then we have an infinite reduction sequence of typable $\lambda^{\mathbf{1},\times,+}$-terms. This is an absurd, since reduction of typable $\lambda^{\mathbf{1},\times,+}$-terms is SN. □

Finally, since λ^{IHLP} is an orthogonal higher-order rewrite system (it has no critical pairs) and is left linear, it is confluent. This follows from standard results in higher-order rewriting [20].

5 Discussion and Related Work

LP through the Curry-de Bruijn-Howard looking glass has already suggested some interesting programming idioms. For example, in [6] a lambda calculus where the reduction history is part of the term is introduced. The following scheme is used to recover subject reduction (which fails for the naive scheme as discussed in Sec. 2), e encoding the derivation of the judgement $\Theta;\Gamma \vdash s \equiv t : A|e$:

$$\frac{\Theta;\Gamma \vdash M^A \mid s \quad \Theta;\Gamma \vdash s \equiv t : A|e}{\Theta;\Gamma \vdash e \rhd M^A \mid s}\,\mathsf{Eq}$$

Strong normalisation is deduced for the resulting term assignment λ^I from weak-normalisation using techniques from higher-order rewriting. Also, a Church-Rosser theorem yields confluence of λ^I. Note that since terms carry information on how a result is computed (very much in line with Lévy labels in rewriting), the CR result may be considered a strengthening of the standard CR result of the typed lambda calculus.

In [10] the history or *computation trail* is allowed to be inspected by introducing *trail variables*; this permits the calculus to model history-based access control [1] and history-based information flow [9]. In that work the following term assignment for $\Box\mathsf{I}$ is proposed, where Δ is a set of trail variables (affine variables that may be read at most once for the purposes of inspecting computation trails):

$$\frac{\Theta;\Delta;\cdot \vdash M^A \mid s \quad \Theta;\Delta;\cdot \vdash s \equiv t : A|e}{\Theta;\Delta';\Gamma \vdash (!^{\Delta}_{e} M)^{[\![t]\!]A} \mid !t}\,\Box\mathsf{I}$$

A term of the form $!^{\Delta}_{e} M$ operates as an audited computation unit, where all computation is audited and *locally scoped* within M.

Also, in [12] by interpreting $\Box A$ as mobile code of type A, LP suggests a calculus of *certified mobile units* which enriches mobile code with certificates (representing type derivations). Such units take the form $\mathsf{box}_s\, M$, s being the certificate and M the executable. Composition of certified mobile units allows one to build mobile code out of other pieces of mobile code *together* with certificates that are also composed out of other certificates. For example, the term

$$\lambda a.\lambda b.unpack\, a\, to\, \langle \overset{\bullet}{u}, \overset{\circ}{u} \rangle\, in\, (unpack\, b\, to\, \langle \overset{\bullet}{v}, \overset{\circ}{v} \rangle\, in\, (box_{\overset{\circ}{u}\cdot\overset{\circ}{v}}\, \overset{\bullet}{u}\, \overset{\bullet}{v}))$$

reads as follows: "Given a mobile unit a and a mobile unit b, extract code $\overset{\bullet}{v}$ and certificate $\overset{\circ}{v}$ from b and extract code $\overset{\bullet}{u}$ and certificate $\overset{\circ}{u}$ from a. Then create new code $\overset{\bullet}{u}\,\overset{\bullet}{v}$ by applying $\overset{\bullet}{u}$ to $\overset{\bullet}{v}$ and a new certificate for this code $\overset{\circ}{u} \cdot \overset{\circ}{v}$. Finally, wrap both of these up into a new mobile unit.". The type system ensures that certificates always correspond to the mobile code with which it is enclosed.

In contrast to [6] this work includes the plus and also explores a more relaxed term assignment (derivation of proof witness equality is not reflected in the term assignment). The reason for relaxing the term assignment is to place the focus of the analysis on the plus, thereby simplifying the terms that it manipulates. That being said and based on current preliminary results, the main role of the plus that suggests itself is its use for typability, as illustrated in Exm. 2.1. It seems to have no run-time effect. However, more work is required in order to gain deeper insight.

6 Conclusions

We study a natural deduction presentation of ILP, an intuitionistic fragment of LP, together with its corresponding term assignment, a variant of those already introduced by the first author and discussed in the previous section. The basic properties of subject reduction, strong normalisation and confluence are easily shown to hold.

We think that a fresh look on realization of IS4 in the setting of IHLP could be an interesting avenue for exploration. It should be noted that this is a non-trivial problem in the presence of inference schemes which mix polarities such as $\supset$ E, hence the reason why the first such proof [2,4] relied on a cut-free sequent calculus presentation of LP. Indeed, all known (to the authors of this work) realization proofs rely on presentations where related [5] occurrences of a $\Box$ do *not* occur both in positive and negative positions. We think it could be interesting to put the well-developed type-inference technology to work *but* to infer the decorations of boxes rather than to infer types. Relations with higher-order unification may appear along the way.

In [7] the so called *Basic Intuitionistic Logic of Proofs* is developed. A modality of the form $[\![u]\!]A$ is introduced, for u a proof variable, and a number of axioms over this modality, together with the axioms of IPC and MP, are shown to capture HA-tautologies. Towards the end of op.cit. operations on proof terms are added and the resulting system is proved to be arithmetically complete in [14]. Developing a proof theory for the latter could be an interesting line of work.

There are numerous proof theoretic approaches to intuitionistic modal logic such as [24,13,17,11], just to name a few. It could be interesting to recast the Logic of Proofs using some of these other approaches rather than the judgemental style adopted here.

Further avenues are those related to the use of natural deduction presentations of fragments of first-order LP. Although first-order LP is not finitely axiomatizable [23,8] (although see [3]), at the cost of losing the connection with Peano Arithmetic, the resulting type theory system could serve as the foundation for a logical framework with decidable forms of reflection. Additionally, we are currently extending our results to full LP, based on classical logic.

References

[1] Abadi, M. and C. Fournet, *Access control based on execution history*, in: *In Proceedings of the 10th Annual Network and Distributed System Security Symposium*, 2003, pp. 107–121.

[2] Artёmov, S., *Operational modal logic*, Technical Report Technical Report MSI 95-29, Cornell University (1995).

[3] Artemov, S. and T. Yavorskaya, *First-order logic of proofs*, Technical report, Technical Report TR–2011005, CUNY Ph. D. Program in Computer Science (2011).

[4] Artёmov, S. N., *Explicit provability and constructive semantics*, Bulletin of Symbolic Logic **7** (2001), pp. 1–36.

[5] Artemov, S. N. and L. D. Beklemishev, *Provability logic*, in: *Handbook of Philosophical Logic, 2nd ed*, 13 (2005), pp. 189–360.

[5] See notion of “family” in [4].

[6] Artëmov, S. N. and E. Bonelli, *The intensional lambda calculus*, in: S. N. Artëmov and A. Nerode, editors, *LFCS*, Lecture Notes in Computer Science **4514** (2007), pp. 12–25.

[7] Artëmov, S. N. and R. Iemhoff, *The basic intuitionistic logic of proofs*, J. Symb. Log. **72** (2007), pp. 439–451.

[8] Artemov, S. N. and T. Yavorskaya (Sidon), *On first order logic of proofs*, Moscow Mathematical Journal **1** (2001), pp. 475–490.

[9] Banerjee, A. and D. A. Naumann, *History-based access control and secure information flow*, in: *Construction and Analysis of Safe, Secure, and Interoperable Smart Devices, International Workshop (CASSIS 2004), Revised Selected Papers, volume 3362 of Lecture Notes in Computer Science* (2005), pp. 27–48.

[10] Bavera, F. and E. Bonelli, *Justification logic and history based computation*, in: A. Cavalcanti, D. Déharbe, M.-C. Gaudel and J. Woodcock, editors, *ICTAC*, Lecture Notes in Computer Science **6255** (2010), pp. 337–351.

[11] Bellin, G., E. Ritter and V. de Paiva, *Extended curry-howard correspondence for a basic constructive modal logic*, in: *Proceedings of Methods for Modalities*, 2001.

[12] Bonelli, E. and F. Feller, *Justification logic as a foundation for certifying mobile computation*, Ann. Pure Appl. Logic **163** (2012), pp. 935–950.

[13] Borghuis, T. J., "Coming to Terms with Modal Logic: On the interpretation of modalities in typed λ-calculus," Ph.D. thesis, Technical University of Eindhoven (1994).

[14] Dashkov, E., *Arithmetical completeness of the intuitionistic logic of proofs*, J. Log. Comput. **21** (2011), pp. 665–682.

[15] Davies, R. and F. Pfenning, *A modal analysis of staged computation*, in: H.-J. Boehm and G. L. S. Jr., editors, *POPL* (1996), pp. 258–270.

[16] Davies, R. and F. Pfenning, *A modal analysis of staged computation*, J. ACM **48** (2001), pp. 555–604.

[17] Ghani, N., V. de Paiva and E. Ritter, *Explicit substitutions for constructive necessity*, in: K. G. Larsen, S. Skyum and G. Winskel, editors, *ICALP*, Lecture Notes in Computer Science **1443** (1998), pp. 743–754.

[18] Kuznets, R., *A note on the use of sum in the logic of proofs* (2009).

[19] Nanevski, A., F. Pfenning and B. Pientka, *Contextual modal type theory*, ACM Trans. Comput. Log. **9** (2008).

[20] Oostrom, V. v., "Confluence for abstract and higher-order rewriting," Ph.D. thesis, VU University Amsterdam (1994).

[21] Pfenning, F. and R. Davies, *A judgmental reconstruction of modal logic*, Mathematical Structures in Computer Science **11** (2001), pp. 511–540.

[22] Schellinx, H., *Basic proof theory, a.s. troelstra and h. schwichtenberg*, Journal of Logic, Language and Information **7** (1998), pp. 221–223.

[23] Sidon, T., *Nonaxiomatizability of predicate logics of proofs*, Moscow University Mathematics Bulletin **53** (1998), pp. 18–22.

[24] Simpson, A., "The Proof Theory and Semantics of Intuitionistic Modal Logic," Ph.D. thesis, Univeristy of Edinburgh (1994).

[25] Steren, G. and E. Bonelli, *Intuitionistic hypothetical logic of proofs*, Technical report, UBA, UNQ (2013), http://tinyurl.com/ihlp-report.