Monad Translations for Higher-Order Logic

Thomas Traversié
Université Paris-Saclay, CentraleSupélec, MICS, Gif-sur-Yvette, France
Université Paris-Saclay, Inria, CNRS, ENS Paris-Saclay, LMF, Gif-sur-Yvette, France

Abstract

Classical logic can be embedded into intuitionistic logic by inserting double negations in formulas. Several translations generalize this idea by using monad operators instead of double negations. They eliminate particular axioms, for instance the principle of excluded middle or the principle of explosion, and therefore can be used to embed classical logic into intuitionistic logic or intuitionistic logic into minimal logic. Such translations have been defined for first-order logic.

In this paper, we define a translation, parameterized by monad operators, for higher-order logic. In particular, the property that any formula and its translation are equivalent in the presence of the eliminated axiom holds under functional extensionality and propositional extensionality. We apply this translation to embed higher-order classical (respectively intuitionistic) logic into higher-order intuitionistic (respectively minimal) logic. By adapting Friedman's trick, we show that coherent formulas correspond to a constructive fragment of higher-order classical logic, meaning that we can transform classical proofs into intuitionistic proofs without modifying the proven statements.

2012 ACM Subject Classification Theory of computation → Constructive mathematics; Theory of computation → Proof theory; Theory of computation → Higher order logic

Keywords and phrases Higher-order logic, Intuitionistic logic, Kuroda's translation, Monad

Digital Object Identifier 10.4230/LIPIcs.FSCD.2025.34

Acknowledgements I would like to thank Marc Aiguier and Olivier Hermant for their helpful remarks and suggestions.

1 Introduction

Intuitionistic logic extends minimal logic with the principle of explosion, which states that any formula can be derived from a contradiction. In minimal logic, contradictions cannot be used to prove any formula. In that sense, minimal logic controls inconsistencies.

$$\bot \Rightarrow A \qquad \text{(principle of explosion)}$$
$$A \vee \neg A \qquad \text{(principle of excluded middle)}$$

Classical logic extends intuitionistic logic with the principle of excluded middle, which states that for any formula A, either A or its negation $\neg A$ is true. Crucially, intuitionistic logic has the disjunction property–if $A \vee B$ holds then either A holds or B holds–and the witness property–if $\exists x.A$ holds then there is a term t such that $A[x \leftarrow t]$ holds. In that sense, proofs in intuitionistic logic are constructive.

Provability in intuitionistic logic (respectively minimal logic) trivially entails provability in classical logic (respectively intuitionistic logic). Following Barr's theorem [2], classical provability entails intuitionistic provability for geometric formulas–which are formulas that can only be built using conjunctions, infinite disjunctions, and existential quantifiers. Although Barr's theorem originates from topos theory, a syntactic proof can be found in [17].

In the general case, however, classical provability does not entail intuitionistic provability. To get around this issue, many different embeddings of classical logic into intuitionistic logic have been proposed over the past century. These translations $A \mapsto A^*$ satisfy two properties:

(i) if A is provable in classical logic then A^* is provable in intuitionistic logic (soundness),
(ii) A^* and A are equivalent in classical logic (characterization).

One approach to embed classical logic into intuitionistic logic is to define double-negation translations, that is to insert double negations inside formulas. Glivenko [11] showed that, for any formula A that is provable in propositional logic, there exists an intuitionistic proof of its double-negation $\neg\neg A$. Kolmogorov [13], Gödel [12], Gentzen [10], and Kuroda [14] defined double-negation translations for first-order logic. Additionally, Kolmogorov's translation and the Gödel-Gentzen translation embed classical logic into minimal logic.

Friedman [9] developed a transformation of intuitionistic formulas that is parameterized by a formula R. The composition of a double-negation translation and Friedman's translation gives rise to a translation from classical logic to intuitionistic logic, in which the double negation is replaced by the operator $A \mapsto (A \Rightarrow R) \Rightarrow R$. The version of Barr's theorem where we only have finitary geometric formulas–called coherent formulas–can be proved using Friedman's translation [18].

So as to generalize even more the double-negation translations, it is possible to resort to *monad operators* instead of double negations. Monad operators are unary connectives T that satisfy $A \Rightarrow TA$ and $(A \Rightarrow TB) \Rightarrow (TA \Rightarrow TB)$ for any formulas A and B. They are called lax modalities in [1], nuclei in [22] and strong monads in [6], and they originate from propositional lax logic [7], topology and category theory. Monad operators have been used to generalize the Gödel-Gentzen translation [6] and Kuroda's translation [22]. These translations eliminate particular axioms, depending on the chosen monad operator, for example the principle of excluded middle or the principle of explosion. Such generic translations are therefore relevant to embed classical logic into intuitionistic logic or intuitionistic logic into minimal logic.

The well-known double-negation translations and their generalizations with monad operators have been defined for *first-order logic*. Brown and Rizkallah [3] showed that Kolmogorov's translation and the Gödel-Gentzen translation cannot be naturally extended to higher-order logic, and they proved that Kuroda's translation can be extended so that the soundness property holds. When we assume both functional extensionality and propositional extensionality, the characterization property also holds [21].

Contribution. In this paper, we define a Kuroda-style monad translation for higher-order logic. Depending on the choice of the monad operator, this translation either embeds higher-order classical logic into higher-order intuitionistic logic or embeds higher-order intuitionistic logic into higher-order minimal logic.

Like the extension of double-negation translation to higher-order logic [21], the characterization property holds under functional extensionality and propositional extensionality. Moreover, we underscore conditions under which the soundness property holds in the presence of both functional extensionality and propositional extensionality.

We refine the monad translation for factorizable monad operators, that are monad operators that satisfy $(TA \Rightarrow TB) \Rightarrow T(A \Rightarrow B)$. The factorizable monad translation directly abstracts Kuroda's translation with factorizable monad operators instead of double negations, and it introduces less monad operators in the formulas than the monad translation. It can be used to embed higher-order classical logic into higher-order intuitionistic logic.

When considering higher-order coherent formulas, the monad translation can be used to show that classical provability entails intuitionistic provability and that intuitionistic provability entails minimal provability. In particular, we are able to constructivize any classical proof of a statement that only involves higher-order coherent formulas.

Outline. We recall in Section 2 the necessary preliminaries about higher-order logic and monad operators. The monad translation for higher-order logic is defined in Section 3. In Section 4, we show that this translation satisfies the soundness and characterization properties, and we examine its behavior in the presence of equality. We illustrate how the monad translation can be used with different monad operators to define embeddings between logics in Section 5, and we study the particular case of higher-order coherent formulas in Section 6. In Section 7, we refine the translation for factorizable monad operators.

2 Preliminaries

The syntax and inference rules of higher-order logic are recalled in Section 2.1. The basic notions about monad operators are summarized in Section 2.2.

2.1 Higher-Order Logic

Higher-order logic is modeled using Church's simple type theory [4]. Types are defined inductively: ι is the type of individuals, o is the type of propositions, and if τ and σ are types then $\tau \to \sigma$ is a type. For every type τ, let $\mathcal{V}_\tau$ be the set of variables of type τ and $\mathcal{C}_\tau$ be a set of constants of type τ. The set of variables $\mathcal{V} := \bigcup_\tau \mathcal{V}_\tau$ and the set of constants $\mathcal{C} := \bigcup_\tau \mathcal{C}_\tau$ are assumed to be disjoint. For any set of constants $\mathcal{C}$, the sets $\Lambda^{\mathcal{C}}_\tau$ of terms of type τ are defined by induction:

- For every $x \in \mathcal{V}_\tau$, $x \in \Lambda^{\mathcal{C}}_\tau$.
- For every $c \in \mathcal{C}_\tau$, $c \in \Lambda^{\mathcal{C}}_\tau$.
- For every $x \in \mathcal{V}_\tau$ and $t \in \Lambda^{\mathcal{C}}_\sigma$, then $(\lambda x.\ t) \in \Lambda^{\mathcal{C}}_{\tau\to\sigma}$.
- For every $t \in \Lambda^{\mathcal{C}}_{\tau\to\sigma}$ and $u \in \Lambda^{\mathcal{C}}_\tau$, then $(t\ u) \in \Lambda^{\mathcal{C}}_\sigma$.

$\lambda x.\ t$ is a λ-abstraction and $t\ u$ is an application. Computation is introduced in this λ-calculus thanks to the β-reduction rule $(\lambda x.\ t)\ u \hookrightarrow t[x \leftarrow u]$, where $t[x \leftarrow u]$ corresponds to the term t in which x has been substituted by u. We denote $\equiv_\beta$ the congruence generated by β-reduction. We write $FV(t_1, \ldots, t_n)$ for the set of free variables that occur in the terms $t_1, \ldots, t_n$.

Formulas are terms of type o. There are particular constants defining the logical connectives and quantifiers: contradiction $\bot$ of type o, implication $\Rightarrow$, conjunction $\wedge$ and disjunction $\vee$ of type $o \to o \to o$, and quantifiers $\forall_\tau$ and $\exists_\tau$ of type $(\tau \to o) \to o$. For convenience, terms of the form $\forall_\tau\ (\lambda x.\ A)$ and $\exists_\tau\ (\lambda x.\ A)$ are simply abbreviated as $\forall x.A$ and $\exists x.A$. The negation is defined by $\neg A := A \Rightarrow \bot$ and the logical biconditional $\Leftrightarrow$ is defined by $A \Leftrightarrow B := (A \Rightarrow B) \wedge (B \Rightarrow A)$. Contexts Γ are finite sequences of formulas.

In the rest of this paper, we consider a logic L which is either minimal logic (ML), intuitionistic logic (IL), or classical logic (CL). The natural deduction rules for CL are given in Figure 1. In IL, we do not consider the principle of excluded middle PEM, and in ML we neither consider PEM nor the principle of explosion Bot-E. We write $\Gamma \vdash_{\mathsf{L}} A$ when $\Gamma \vdash A$ is derivable in the logic L.

We can also define, for every type τ, an equality symbol $=_\tau$ of type $\tau \to \tau \to o$. The symbols are infix, and we write $t = u$ when there is no ambiguity on the type τ. The natural deduction rules for equality are given in Figure 2. We write $\Gamma \vdash^*_{\mathsf{L}} A$ with $* \in \{\mathfrak{e}, \mathfrak{ep}, \mathfrak{ef}, \mathfrak{efp}\}$ when $\Gamma \vdash_{\mathsf{L}} A$ is derivable with additional inference rules: with Eq-I and Eq-E when $\mathfrak{e}$ occurs in $*$, with PropExt when $\mathfrak{p}$ occurs in $*$, and with FunExt when $\mathfrak{f}$ occurs in $*$.

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}\ \textsc{Imp-I} \qquad \frac{\Gamma \vdash A \Rightarrow B \qquad \Gamma \vdash A}{\Gamma \vdash B}\ \textsc{Imp-E}$$

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \textsc{And-I} \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ \textsc{And-EL} \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\ \textsc{And-ER}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ \textsc{Or-IL} \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ \textsc{Or-IR} \qquad \frac{\Gamma \vdash A \vee B \qquad \Gamma, A \vdash C \qquad \Gamma, B \vdash C}{\Gamma \vdash C}\ \textsc{Or-E}$$

$$\frac{\Gamma \vdash A \qquad x \notin FV(\Gamma)}{\Gamma \vdash \forall x.A}\ \textsc{All-I} \qquad \frac{\Gamma \vdash \forall x.A}{\Gamma \vdash A[x \leftarrow t]}\ \textsc{All-E}$$

$$\frac{\Gamma \vdash A[x \leftarrow t]}{\Gamma \vdash \exists x.A}\ \textsc{Ex-I} \qquad \frac{\Gamma \vdash \exists x.A \qquad \Gamma, A \vdash C \qquad x \notin FV(\Gamma, C)}{\Gamma \vdash C}\ \textsc{Ex-E}$$

$$\frac{}{\Gamma, A, \Delta \vdash A}\ \textsc{Ax} \qquad \frac{\Gamma \vdash A \qquad A \equiv_\beta B}{\Gamma \vdash B}\ \textsc{Conv}$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash A}\ \textsc{Bot-E} \qquad \frac{}{\Gamma \vdash A \vee \neg A}\ \text{PEM}$$

Figure 1 Natural deduction rules for classical logic.

$$\frac{}{\Gamma \vdash u = u}\ \textsc{Eq-I} \qquad \frac{\Gamma \vdash A[x \leftarrow u] \qquad \Gamma \vdash u = v}{\Gamma \vdash A[x \leftarrow v]}\ \textsc{Eq-E}$$

$$\frac{\Gamma \vdash f\ x = g\ x \qquad x \notin FV(\Gamma, f, g)}{\Gamma \vdash f = g}\ \textsc{FunExt} \qquad \frac{\Gamma \vdash A \Rightarrow B \qquad \Gamma \vdash B \Rightarrow A}{\Gamma \vdash A = B}\ \textsc{PropExt}$$

Figure 2 Natural deduction rules for equality.

2.2 Monad Operators

We say that T is a unary connective when it is a closed term of type $o \to o$. A unary connective T is a monad operator of L when the judgments (unit) and (bind) hold for any formulas A and B.

$$\vdash_{\mathsf{L}} A \Rightarrow TA \tag{unit}$$

$$\vdash_{\mathsf{L}} (A \Rightarrow TB) \Rightarrow (TA \Rightarrow TB) \tag{bind}$$

We use the terminology "monad operator" to avoid any confusion with monads in the sense of category theory. Monad operators are called lax modalities in [1] and nuclei in [22]. This definition of monad operators is equivalent to the axiomatization of strong monads in [6]. Following the Curry-Howard correspondence, monad operators coincide with the notion of monad in programming languages [15], for instance in Haskell.

▶ **Example 1.** There are many monad operators of minimal logic:

- $TA := A \vee \bot$,
- $T_R A := (A \Rightarrow R) \Rightarrow R$, with parameter R, which corresponds to the continuation monad,
- $TA := \neg\neg A$, which is a special case of the previous one with $R := \bot$,
- $T_R A := (A \Rightarrow R) \Rightarrow A$, with parameter R, which corresponds to the Peirce monad [6].

We recall some formulas about monad operators that will be used in the rest of the paper.

▶ **Proposition 2.** *Let T be a monad operator of* L, *and A and B be formulas.*

1. $\vdash_{\mathsf{L}} TTA \Leftrightarrow TA$
2. $\vdash_{\mathsf{L}} (A \Rightarrow B) \Rightarrow (TA \Rightarrow TB)$
3. $\vdash_{\mathsf{L}} T(A \wedge B) \Leftrightarrow (TA \wedge TB)$
4. $\vdash_{\mathsf{L}} T(A \Rightarrow B) \Rightarrow (TA \Rightarrow TB)$
5. $\vdash_{\mathsf{L}} T(A \Rightarrow TB) \Leftrightarrow (TA \Rightarrow TB)$
6. $\vdash_{\mathsf{L}} T(A \vee B) \Leftrightarrow T(TA \vee TB)$
7. $\vdash_{\mathsf{L}} T(\forall x.TA) \Leftrightarrow \forall x.TA$
8. $\vdash_{\mathsf{L}} T(\exists x.TA) \Leftrightarrow T(\exists x.A)$

Proof. We only show the most interesting cases.

- **Item 3 ($\Rightarrow$)**: We have $\vdash_{\mathsf{L}} (A \wedge B) \Rightarrow A$ and $\vdash_{\mathsf{L}} (A \wedge B) \Rightarrow B$. Using Item 2 we easily derive $\vdash_{\mathsf{L}} T(A \wedge B) \Rightarrow (TA \wedge TB)$.
- **Item 3 ($\Leftarrow$)**: Using (bind), we directly have $\vdash_{\mathsf{L}} (A \Rightarrow T(A \wedge B)) \Rightarrow TA \Rightarrow T(A \wedge B)$ and $\vdash_{\mathsf{L}} (B \Rightarrow T(A \wedge B)) \Rightarrow TB \Rightarrow T(A \wedge B)$. Using these two facts and (unit), we easily derive $\vdash_{\mathsf{L}} TA \Rightarrow TB \Rightarrow T(A \wedge B)$. We conclude $\vdash_{\mathsf{L}} (TA \wedge TB) \Rightarrow T(A \wedge B)$.
- **Item 4**: We know that $\vdash_{\mathsf{L}} ((A \Rightarrow B) \wedge A) \Rightarrow B$. We derive $\vdash_{\mathsf{L}} T((A \Rightarrow B) \wedge A) \Rightarrow TB$ using Item 2. By Item 3, we get $\vdash_{\mathsf{L}} (T(A \Rightarrow B) \wedge TA) \Rightarrow TB$, therefore we conclude $\vdash_{\mathsf{L}} T(A \Rightarrow B) \Rightarrow (TA \Rightarrow TB)$.
- **Item 5 ($\Rightarrow$)**: We have $\vdash_{\mathsf{L}} T(A \Rightarrow TB) \Rightarrow (TA \Rightarrow TTB)$ using Item 4. We conclude $\vdash_{\mathsf{L}} T(A \Rightarrow TB) \Rightarrow (TA \Rightarrow TB)$ using Item 1.
- **Item 5 ($\Leftarrow$)**: Using (unit), we derive $\vdash_{\mathsf{L}} (TA \Rightarrow TB) \Rightarrow (A \Rightarrow TB)$ and $\vdash_{\mathsf{L}} (A \Rightarrow TB) \Rightarrow T(A \Rightarrow TB)$. It follows $\vdash_{\mathsf{L}} (TA \Rightarrow TB) \Rightarrow T(A \Rightarrow TB)$.
- **Item 6 ($\Leftarrow$)**: Using (bind), we get $\vdash_{\mathsf{L}} (A \Rightarrow T(A \vee B)) \Rightarrow (TA \Rightarrow T(A \vee B))$ and $\vdash_{\mathsf{L}} (B \Rightarrow T(A \vee B)) \Rightarrow (TB \Rightarrow T(A \vee B))$. Using (unit), we derive $\vdash_{\mathsf{L}} TA \Rightarrow T(A \vee B)$ and $\vdash_{\mathsf{L}} TB \Rightarrow T(A \vee B)$. It follows $\vdash_{\mathsf{L}} (TA \vee TB) \Rightarrow T(A \vee B)$. We conclude using (bind).
- **Item 7 ($\Rightarrow$)**: We easily show $\vdash_{\mathsf{L}} (\forall x.TA) \Rightarrow TA[x \leftarrow y]$ for a fresh variable y. Using (bind) we get $\vdash_{\mathsf{L}} T(\forall x.TA) \Rightarrow TA[x \leftarrow y]$. We derive $T(\forall x.TA) \vdash_{\mathsf{L}} TA[x \leftarrow y]$ and therefore $T(\forall x.TA) \vdash_{\mathsf{L}} \forall x.TA$. We conclude $\vdash_{\mathsf{L}} T(\forall x.TA) \Rightarrow \forall x.TA$. ◀

3 Monad Translation

The standard double-negation translations [13, 12, 10, 14] have been defined for first-order logic. Several translations generalize them to monad operators [1, 6, 22] and extend them to higher-order logic [3, 21]. We want to define a translation that does both.

Kolmogorov's translation [13] and the Gödel-Gentzen translation [12, 10] cannot be extended to higher-order logic directly [3], because they do not preserve β-conversion. For example, let us take the Gödel-Gentzen translation of $(\lambda R.\ R \wedge Q)\ P \hookrightarrow P \wedge Q$. The translation of $(\lambda R.\ R \wedge Q)\ P$ is $(\lambda R.\ \neg\neg R \wedge \neg\neg Q)\ \neg\neg P$, which β-reduces to $\neg\neg\neg\neg P \wedge \neg\neg Q$, while the translation of $P \wedge Q$ is $\neg\neg P \wedge \neg\neg Q$. Unlike Kolmogorov's translation and the Gödel-Gentzen translation, Kuroda's translation [14] preserves β-conversion and can be directly extended to higher-order logic [3, 21].

Moreover, a Kuroda-style monad translation [22] has been defined for first-order logic. It corresponds to Kuroda's translation, in which we insert monad operators instead of double negations. It also introduces additional monad operators after implications, and such modification is necessary to generalize Kuroda's translation to *any* monad operator. We will see in Section 7 how to remove this constraint for *particular* monad operators.

We define here a Kuroda-style monad translation for higher-order logic, taking advantage of the ideas developed in [21] and [22].

▶ **Definition 3** (Monad translation for higher-order logic). *Let A be a formula in higher-order logic and T be a monad operator. Its monad translation is $A^T := TA_T$, where $t \mapsto t_T$ is inductively defined by:*

$$\begin{array}{lcl} x_T & := & x \\ c_T & := & \begin{cases} \lambda p.\ \forall x. T(p\ x) & \textit{if } c = \forall \\ \lambda p.\ \lambda q.\ p \Rightarrow Tq & \textit{if } c = \Rightarrow \\ c & \textit{otherwise} \end{cases} \\ (\lambda x.\ t)_T & := & \lambda x.\ t_T \\ (t\ u)_T & := & t_T\ u_T \end{array}$$

In first-order logic, we have $(A[z \leftarrow w])^T = A^T[z \leftarrow w]$. As we are in higher-order logic, the substituted term w may be modified by the translation. We would intuitively expect $(A[z \leftarrow w])^T = A^T[z \leftarrow w^T]$, but we actually have $(A[z \leftarrow w])^T = A^T[z \leftarrow w_T]$, because the monad operator in front of the formula is inserted after the inductive translation. It follows that the translation $t \mapsto t_T$ commutes with substitution, but not $A \mapsto A^T$. Both translations $t \mapsto t_T$ and $A \mapsto A^T$ preserve convertibility.

▶ **Proposition 4.** *For any term t, we have $(t[z \leftarrow w])_T = t_T[z \leftarrow w_T]$.*

Proof. By induction on the term t. ◀

▶ **Corollary 5.** *For any higher-order formula A, we have $(A[z \leftarrow w])^T = A^T[z \leftarrow w_T]$.*

In higher-order logic, terms are considered modulo β-convertibility. It follows that the translation must preserve β-convertibility for it to preserve provability.

▶ **Proposition 6.** *For any terms t and u, if $t \equiv_\beta u$ then $t_T \equiv_\beta u_T$.*

Proof. By definition $((\lambda x.\ t)\ u)_T = (\lambda x.\ t_T)\ u_T$, and we have $(\lambda x.\ t_T)\ u_T \hookrightarrow t_T[x \leftarrow u_T]$. Using Proposition 4, we get $((\lambda x.\ t)\ u)_T \equiv_\beta (t[x \leftarrow u])_T$. Closure by context, reflexivity, symmetry, and transitivity are immediate. ◀

▶ **Corollary 7.** *For any higher-order formulas A and B, if $A \equiv_\beta B$ then $A^T \equiv_\beta B^T$.*

4 Monad Embedding

The monad translation generalizes double-negation translations with monad operators instead of double negations. Likewise, the monad translation eliminates the use of a particular formula [1, 6], called T-elimination, while double-negation translations eliminate the use of the principle of excluded middle.

$$TA \Rightarrow A \qquad (T\text{-elimination})$$

More formally, we extend the logic L with the inference rule T-Elim, and we write $\Gamma \vdash_{\mathsf{L+T}} A$ when $\Gamma \vdash A$ is derivable using the rules of L and T-Elim.

$$\frac{}{\Gamma \vdash TA \Rightarrow A}\ \text{T-Elim}$$

The monad translation eliminates any use of the T-Elim inference rule–this is the soundness property. The characterization property states that any formula and its translation are equivalent assuming T-Elim.

When extending the double-negation elimination to higher-order logic [21], the characterization property does not trivially hold, but does hold when we consider functional extensionality and propositional extensionality. The same reasoning applies for the extension of the monad translation to higher-order logic.

▶ **Theorem 8** (Monad embedding). *Let T be a monad operator of L such that we have $\vdash_{\mathsf{L}} (TA)_T \Leftrightarrow TA_T$ for any formula A. Let Γ, A be higher-order formulas.*

1. *If $\Gamma \vdash_{\mathsf{L+T}} A$ then $\Gamma_T \vdash_{\mathsf{L}} A^T$.*
2. *$\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} A^T \Leftrightarrow A$.*

▶ Remark 9. For Kolmogorov's translation and the Gödel-Gentzen translation, the soundness property is stated with Γ^T instead of Γ_T. When considering Kuroda's translation, we can actually omit the monad operators at the head of the formulas of the context. The version with Γ^T is provable as well.

Proof of Theorem 8(1). We proceed by induction on the derivation. Most of the cases are direct applications of Proposition 2. Conv derives from Corollary 7. We only show the most interesting cases:

- Or-E: Suppose that $\Gamma_T \vdash_{\mathsf{L}} T(A_T \vee B_T)$ and $\Gamma_T, A_T \vdash_{\mathsf{L}} TC_T$ and $\Gamma_T, B_T \vdash_{\mathsf{L}} TC_T$. To prove $\Gamma_T \vdash_{\mathsf{L}} TC_T$, it suffices to have $\Gamma_T \vdash_{\mathsf{L}} T(A_T \vee B_T) \Rightarrow TC_T$.
  Using the second and third hypotheses, we derive $\Gamma_T \vdash_{\mathsf{L}} (A_T \vee B_T) \Rightarrow TC_T$. Using (bind) we get $\Gamma_T \vdash_{\mathsf{L}} T(A_T \vee B_T) \Rightarrow TC_T$ and we conclude the case.
- All-I: Suppose $\Gamma_T \vdash_{\mathsf{L}} TA_T$. We derive $\Gamma_T \vdash_{\mathsf{L}} \forall x.TA^T$, that is $\Gamma_T \vdash_{\mathsf{L}} (\forall x.A)_T$. We conclude using (unit).
- All-E: Suppose $\Gamma_T \vdash_{\mathsf{L}} T(\forall x.TA_T)$. Using Proposition 2(7), we get $\Gamma_T \vdash_{\mathsf{L}} \forall x.TA_T$. From All-E we obtain $\Gamma_T \vdash_{\mathsf{L}} (TA_T)[x \leftarrow t_T]$. We conclude with Corollary 5.
- Ex-I: Suppose $\Gamma_T \vdash_{\mathsf{L}} T(A[x \leftarrow t])_T$. Using Corollary 5, we get $\Gamma_T \vdash_{\mathsf{L}} TA_T[x \leftarrow t_T]$. From Ex-I we obtain $\Gamma_T \vdash_{\mathsf{L}} \exists x.TA_T$. We conclude using (unit) and Proposition 2(8).
- Ex-E: Suppose $\Gamma_T \vdash_{\mathsf{L}} T(\exists x.A_T)$ and $\Gamma_T, A_T \vdash_{\mathsf{L}} TC_T$. To conclude the proof, it suffices to have $\Gamma_T \vdash_{\mathsf{L}} T(\exists x.A_T) \Rightarrow TC_T$.
  We have $\Gamma_T, \exists x.A_T \vdash_{\mathsf{L}} \exists x.A_T$ (by Ax) and $\Gamma_T, \exists x.A_T, A_T \vdash_{\mathsf{L}} TC_T$ (by weakening of the second hypothesis). Using Ex-E, we derive $\Gamma_T, \exists x.A_T \vdash_{\mathsf{L}} TC_T$ and therefore $\Gamma_T \vdash_{\mathsf{L}} (\exists x.A_T) \Rightarrow TC_T$. Using (bind), we obtain $\Gamma_T \vdash_{\mathsf{L}} T(\exists x.A_T) \Rightarrow TC_T$ and we conclude the case.
- Bot-E: Suppose $\Gamma_T \vdash_{\mathsf{IL}} T\bot$. As we are in intuitionistic logic, we have $\Gamma_T \vdash_{\mathsf{IL}} \bot \Rightarrow A_T$. We deduce $\Gamma_T \vdash_{\mathsf{IL}} T\bot \Rightarrow TA_T$ using Proposition 2(2). We conclude $\Gamma_T \vdash_{\mathsf{IL}} TA_T$.
- PEM: As we are in classical logic, we have $\Gamma_T \vdash_{\mathsf{CL}} A_T \vee (A_T \Rightarrow \bot)$. Using (unit), we derive $\Gamma_T \vdash_{\mathsf{CL}} A_T \vee (A_T \Rightarrow T\bot)$ and then we conclude $\Gamma_T \vdash_{\mathsf{CL}} T(A_T \vee (A_T \Rightarrow T\bot))$.
- T-Elim: We have $(TA \Rightarrow A)_T = ((TA)_T \Rightarrow TA_T)$. We know $\vdash_{\mathsf{L}} (TA)_T \Leftrightarrow TA_T$, so we derive $\Gamma_T \vdash_{\mathsf{L}} (TA \Rightarrow A)_T$. We conclude $\Gamma_T \vdash_{\mathsf{L}} T(TA \Rightarrow A)_T$ using (unit). ◀

Before proving the second item of Theorem 8, we show the following lemma, stating that any term and its translation are equal, assuming T-Elim, functional extensionality and propositional extensionality.

▶ **Lemma 10.** *For any term t, we have $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} t_T = t$.*

Proof. We proceed by induction on the term t. We directly have $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} x_T = x$ and $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} c_T = c$ for $c \notin \{\forall, \Rightarrow\}$. For $c \in \{\forall, \Rightarrow\}$, we derive $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} c_T = c$ from PropExt, FunExt, T-Elim and (unit). We have $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} (t\ u)_T = t\ u$ using the induction hypotheses and Eq-E. We derive $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} (\lambda x.\ t)_T = \lambda x.\ t$ using the induction hypothesis and FunExt. ◀

Proof of Theorem 8(2). We derive $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} A_T \Leftrightarrow A$ using Lemma 10, and then we conclude $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} A^T \Leftrightarrow A$ using (unit) and T-Elim. ◀

We proved that the monad translation satisfies the soundness and characterization properties. For the characterization property, we had to introduce symbols of equality and to assume both functional extensionality and propositional extensionality. But does the soundness property still hold in the presence of equality?

The proof of the soundness property does not naturally extend when we consider the symbols and inference rules of equality. Let us consider the formulas

$$\forall x \forall y.\ T(x = y) \Rightarrow x = y \qquad (\Delta^{\mathfrak{ef}})$$

$$\forall x \forall y.\ (Tx = Ty) \Rightarrow x = y \qquad (\Delta^{\mathfrak{ep}})$$

regarding T and equality predicates. We write $\Delta^{\mathfrak{efp}}$ for the context $\Delta^{\mathfrak{ef}}, \Delta^{\mathfrak{ep}}$. When proving the soundness property in the presence of equality, the first formula $\Delta^{\mathfrak{ef}}$ is useful for the FunExt case and the second formula $\Delta^{\mathfrak{ep}}$ is useful for the PropExt case.

▶ **Theorem 11** (Soundness for equality)**.** *Let T be a monad operator of L such that we have $\vdash_{\mathsf{L}} (TA)_T \Leftrightarrow TA_T$ for any formula A. Let Γ, A be higher-order formulas.*

1. *If $\Gamma \vdash^{\mathfrak{e}}_{\mathsf{L+T}} A$ then $\Gamma_T \vdash^{\mathfrak{e}}_{\mathsf{L}} A^T$.*
2. *For $* \in \{\mathfrak{ef}, \mathfrak{ep}, \mathfrak{efp}\}$, if $\Gamma \vdash^{*}_{\mathsf{L+T}} A$ then $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} A^T$.*

Proof. For the first item, we complete the proof of Theorem 8 with the following cases:

- Eq-I: We directly use Eq-I and (unit).
- Eq-E: Suppose $\Gamma_T \vdash^{*}_{\mathsf{L}} TA_T[x \leftarrow u_T]$ and $\Gamma_T \vdash^{*}_{\mathsf{L}} T(u_T = v_T)$. Using Eq-E, we derive $\vdash^{*}_{\mathsf{L}} A_T[x \leftarrow u_T] \Rightarrow (u_T = v_T) \Rightarrow A_T[x \leftarrow v_T]$. Therefore, we get $\vdash^{*}_{\mathsf{L}} TA_T[x \leftarrow u_T] \Rightarrow T(u_T = v_T) \Rightarrow TA_T[x \leftarrow v_T]$ using Proposition 2(2) and Proposition 2(4). We conclude $\Gamma_T \vdash^{*}_{\mathsf{L}} TA_T[x \leftarrow v_T]$.

For the second item, most cases are those of the first item that are reused using weakening on Δ^*. The remaining cases are:

- FunExt: Suppose $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} T(f_T\ x = g_T\ x)$. Since $x \notin FV(\Gamma, f, g)$, we directly have $x \notin FV(\Gamma_T, f_T, g_T)$. We get $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} f_T\ x = g_T\ x$ using $\Delta^{\mathfrak{ef}}$. We conclude using FunExt and (unit).
- PropExt: Suppose $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} T(A_T \Rightarrow TB_T)$ and $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} T(B_T \Rightarrow TA_T)$. Using Proposition 2(5), we get $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} TA_T \Rightarrow TB_T$ and $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} TB_T \Rightarrow TA_T$. Using PropExt we derive $\Delta^*, \Gamma_T \vdash^{*}_{\mathsf{L}} TA_T = TB_T$. We conclude using $\Delta^{\mathfrak{ep}}$ and (unit). ◀

When extending Kuroda's double-negation translation to higher-order logic [21], we only considered the formula $\Delta^{\mathfrak{ef}}$. The generalization to any monad operator requires us to additionally consider the formula $\Delta^{\mathfrak{ep}}$. In Section 7, we will see that this additional assumption can be dropped when considering factorizable monad operators.

5 Embeddings between Logics

Such monad translation can be directly applied with particular monad operators to embed classical logic into intuitionistic logic or intuitionistic logic into minimal logic.

There are many classical laws, such as the principle of excluded middle, the double-negation elimination, Peirce's law, or Clavius's law. All these laws are equivalent in intuitionistic logic.

$$\neg\neg A \Rightarrow A \qquad \text{(double-negation elimination)}$$
$$(\neg A \Rightarrow A) \Rightarrow A \qquad \text{(Clavius's law)}$$
$$((A \Rightarrow B) \Rightarrow A) \Rightarrow A \qquad \text{(Peirce's law)}$$

The T-elimination principle corresponds to the double-negation elimination when we choose $TA := (A \Rightarrow \bot) \Rightarrow \bot$ and to Clavius's law when we choose $TA := (A \Rightarrow \bot) \Rightarrow A$. For these monad operators, the T-Elim inference rule is therefore equivalent to PEM, and eliminating any use of T-elim in the derivations means transforming classical provability into intuitionistic provability. When $T_R A := (A \Rightarrow R) \Rightarrow A$, the T-elimination principle only corresponds to an instance of Peirce's law with $B := R$.

▶ **Corollary 12** (Embeddings of classical logic into intuitionistic logic)**.** *Let Γ, A be higher-order formulas.*

1. *If $\Gamma \vdash_{\mathsf{CL}} A$ then $\Gamma_T \vdash_{\mathsf{IL}} A^T$ with $TA := (A \Rightarrow \bot) \Rightarrow \bot$.*
2. *If $\Gamma \vdash_{\mathsf{CL}} A$ then $\Gamma_T \vdash_{\mathsf{IL}} A^T$ with $TA := (A \Rightarrow \bot) \Rightarrow A$.*
3. *If $\Gamma \vdash_{\mathsf{CL}} A$ then there exists a formula R such that $\Gamma_{T_R} \vdash_{\mathsf{IL}} A^{T_R}$ with $T_R A := (A \Rightarrow R) \Rightarrow A$.*

Moreover, for each of the above, we have $\vdash^{\mathfrak{cfp}}_{\mathsf{CL}} A^T \Leftrightarrow A$.

Proof. We apply Theorem 8 with $\mathsf{L} := \mathsf{IL}$. T-Elim is equivalent to PEM, so $\mathsf{IL}+\mathsf{T}$ corresponds to CL. For the third item, we have to give the correct parameter R. The derivation $\Gamma \vdash_{\mathsf{CL}} A$ is finite, so Peirce's law is used a finite number of times with formulas $B_1, \dots, B_n$ of free variables $\vec{x_1}, \dots, \vec{x_n}$. Such free variables could be captured later in the derivation. As remarked by Escardó and Oliva [6], $T_{R_1 \wedge R_2} A \Rightarrow A \vdash_{\mathsf{IL}} (T_{R_1} A \Rightarrow A) \wedge (T_{R_2} A \Rightarrow A)$. With the additional remark that $T_{\forall x.R} A \Rightarrow A \vdash_{\mathsf{IL}} T_{R[x \leftarrow t]} A \Rightarrow A$, we define R to be the conjunction $\forall \vec{x_1}.B_1 \wedge \dots \wedge \forall \vec{x_n}.B_n$, so that the monad translation eliminates the n instances of Peirce's law that occur in the derivation. ◀

When $TA := A \vee \bot$, the T-elimination principle is equivalent to the principle of explosion, and the T-Elim inference rule is therefore equivalent to the Bot-E inference rule. In that case, eliminating any use of T-elim in the derivations means transforming intuitionistic provability into minimal provability.

▶ **Corollary 13** (Embedding of intuitionistic logic into minimal logic)**.** *Let Γ, A be higher-order formulas. If $\Gamma \vdash_{\mathsf{IL}} A$ then $\Gamma_T \vdash_{\mathsf{ML}} A^T$ with $TA := A \vee \bot$. Moreover, we have $\vdash^{\mathfrak{cfp}}_{\mathsf{IL}} A^T \Leftrightarrow A$.*

Proof. We apply Theorem 8 with $\mathsf{L} := \mathsf{ML}$. T-Elim is equivalent to Bot-E, so $\mathsf{ML} + \mathsf{T}$ corresponds to IL. ◀

6 Fragment of Coherent Formulas

Coherent formulas are formulas that can only be built using conjunctions, disjunctions and existential quantifiers. Because monad operators are only inserted after universal quantifiers and implications, it follows that $A_T = A$ for any coherent formula A. We lift to higher-order logic two tricks for coherent formulas that rely on this observation.

The first trick [5, 22] allows us to derive minimal provability from intuitionistic provability when the context is empty and when the formula is coherent.

▶ **Corollary 14.** *Let* Γ, A *be higher-order coherent formulas. If* $\vdash_{\mathsf{IL}} A$ *then* $\vdash_{\mathsf{ML}} A$.

Proof. Suppose $\vdash_{\mathsf{IL}} A$. We apply Theorem 8 with $TA := A \vee \bot$, and we derive $\vdash_{\mathsf{ML}} TA_T$, that is $\vdash_{\mathsf{ML}} A \vee \bot$. We know that minimal logic has the disjunction property, and that we cannot prove $\bot$. Hence, we conclude $\vdash_{\mathsf{ML}} A$. ◀

The second trick [18, 22] allows us to derive intuitionistic provability from classical provability when the formulas involved are coherent. It corresponds to a weaker version–with finite disjunctions–of Barr's theorem [2] extended to higher-order logic.

▶ **Theorem 15.** *Let* Γ, A *be higher-order coherent formulas. If* $\Gamma \vdash_{\mathsf{CL}} A$ *then* $\Gamma \vdash_{\mathsf{IL}} A$.

To prove this result, we adapt to higher-order logic Palmgren's idea [18] of using Friedman's translation [9]. As an intermediary lemma, we prove that, for higher-order logic, Friedman's translation transforms classical proofs into intuitionistic proofs. Remark that we cannot directly apply Theorem 8, as the T-elimination principle does not correspond to a classical law when $TA := (A \Rightarrow R) \Rightarrow R$.

▶ **Lemma 16.** *Let* $TA := (A \Rightarrow R) \Rightarrow R$ *with parameter* R. *Let* Γ, A *be higher-order formulas. If* $\Gamma \vdash_{\mathsf{CL}} A$ *then* $\Gamma_T \vdash_{\mathsf{IL}} A^T$.

Proof. We proceed by induction on the derivation. All the cases are similar to the cases of Theorem 8, except that we do not need the T-Elim case anymore, and that we change the PEM case. We prove $\Gamma_T \vdash_{\mathsf{IL}} T(A \vee \neg A)_T$ using

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma_T, B \Rightarrow R, A_T, \bot \Rightarrow R \vdash_{\mathsf{IL}} A_T}\ \textsc{Ax}}{\Gamma_T, B \Rightarrow R, A_T, \bot \Rightarrow R \vdash_{\mathsf{IL}} B}\ \textsc{Or-IL}}{\Gamma_T, B \Rightarrow R, A_T, \bot \Rightarrow R \vdash_{\mathsf{IL}} R}\ \textsc{Imp-E} \text{ with } B \Rightarrow R}{\Gamma_T, B \Rightarrow R \vdash_{\mathsf{IL}} A_T \Rightarrow ((\bot \Rightarrow R) \Rightarrow R)}\ \textsc{Imp-I} \times 2}{\Gamma_T, B \Rightarrow R \vdash_{\mathsf{IL}} B}\ \textsc{Or-IR}}{\Gamma_T, B \Rightarrow R \vdash_{\mathsf{IL}} R}\ \textsc{Imp-E} \text{ with } B \Rightarrow R}{\Gamma_T \vdash_{\mathsf{IL}} ((A \vee (A \Rightarrow ((\bot \Rightarrow R) \Rightarrow R))) \Rightarrow R) \Rightarrow R}\ \textsc{Imp-I}$$

where B is an abbreviation for the formula $A_T \vee (A_T \Rightarrow ((\bot \Rightarrow R) \Rightarrow R))$. ◀

Proof of Theorem 15. Suppose $\Gamma \vdash_{\mathsf{CL}} A$. Without loss of generality, we assume that Γ and A have no free variables, as we can replace them by fresh constants. We apply Lemma 16, and we derive $\Gamma_{T_R} \vdash_{\mathsf{IL}} T_R A_{T_R}$, that is $\Gamma \vdash_{\mathsf{IL}} (A \Rightarrow R) \Rightarrow R$. Choosing $R := A$, we get $\Gamma \vdash_{\mathsf{IL}} A$. ◀

In other words, coherent formulas correspond to a constructive fragment of higher-order classical logic. Constructive fragments of classical logic have been studied for propositional logic [11, 12] and first-order logic [19]. This is the first time, to our knowledge, that such results are proved for higher-order logic.

7 Refinement for Factorizable Monad Operators

So far, we have generalized Kuroda's translation to *any* monad operator, at the expense of the insertion of additional monad operators after implications. In this section, we refine the translation for *factorizable* monad operators. Monad operators of L are factorizable when they satisfy the (factorization) judgment for any propositions A and B.

$$\vdash_{\mathsf{L}} (TA \Rightarrow TB) \Rightarrow T(A \Rightarrow B) \qquad \text{(factorization)}$$

For instance, (factorization) holds for $T_R A := (A \Rightarrow R) \Rightarrow A$ in minimal logic and for $TA := (A \Rightarrow \bot) \Rightarrow \bot$ in intuitionistic logic.

7.1 Factorizable Monad Translation

For factorizable monad operators, we do not need to insert a monad operator in the implication case of the translation [22]. In that sense, the factorizable monad translation is a direct abstraction of Kuroda's translation, in which double negations have been replaced by factorizable monad operators.

▶ **Definition 17** (Factorizable monad translation for higher-order logic). *Let A be a formula in higher-order logic and T be a factorizable monad operator. Its factorizable monad translation is $A^{\widetilde{T}} := TA_{\widetilde{T}}$, where $t \mapsto t_{\widetilde{T}}$ is inductively defined by:*

$$
\begin{array}{lll}
x_{\widetilde{T}} & := & x \\
c_{\widetilde{T}} & := & \begin{cases} \lambda p.\ \forall x. T(p\ x) & \textit{if } c = \forall \\ c & \textit{otherwise} \end{cases} \\
(\lambda x.\ t)_{\widetilde{T}} & := & \lambda x.\ t_{\widetilde{T}} \\
(t\ u)_{\widetilde{T}} & := & t_{\widetilde{T}}\ u_{\widetilde{T}}
\end{array}
$$

The translation $t \mapsto t_{\widetilde{T}}$ and $A \mapsto A^{\widetilde{T}}$ satisfy the same properties as $t \mapsto t_T$ and $A \mapsto A^T$ concerning substitution and convertibility.

▶ **Proposition 18.** *Let t and u be terms, and A and B be higher-order formulas.*

1. $(t[z \leftarrow w])_{\widetilde{T}} = t_{\widetilde{T}}[z \leftarrow w_{\widetilde{T}}]$.
2. $(A[z \leftarrow w])^{\widetilde{T}} = A^{\widetilde{T}}[z \leftarrow w_{\widetilde{T}}]$.
3. *If $t \equiv_\beta u$ then $t_{\widetilde{T}} \equiv_\beta u_{\widetilde{T}}$.*
4. *If $A \equiv_\beta B$ then $A^{\widetilde{T}} \equiv_\beta B^{\widetilde{T}}$.*

Proof. We adapt the proofs of Proposition 4, Corollary 5, Proposition 6 and Corollary 7. ◀

7.2 Factorizable Monad Embedding

The factorizable monad translation satisfies the soundness and characterization properties. It only applies to factorizable monad operators, but it simplifies the monad translation, as it introduces fewer monad operators while satisfying the same result.

▶ **Theorem 19** (Factorizable monad embedding). *Let T be a factorizable monad operator of L such that $\vdash_{\mathsf{L}} (TA)_{\widetilde{T}} \Leftrightarrow TA_{\widetilde{T}}$ for any formula A. Let Γ, A be higher-order formulas.*

1. *If $\Gamma \vdash_{\mathsf{L+T}} A$ then $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} A^{\widetilde{T}}$.*
2. $\vdash^{\mathfrak{cfp}}_{\mathsf{L+T}} A^{\widetilde{T}} \Leftrightarrow A$.

Proof. We prove the first item by induction on the derivation. Most of the cases are the same as for Theorem 8. <u>Conv</u> derives from Proposition 18. We only show the proof of the cases that differ:

- <u>Imp-I</u>: Suppose that $\Gamma_{\widetilde{T}}, A_{\widetilde{T}} \vdash_{\mathsf{L}} TB_{\widetilde{T}}$. We get $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} A_{\widetilde{T}} \Rightarrow TB_{\widetilde{T}}$, and we derive $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} TA_{\widetilde{T}} \Rightarrow TB_{\widetilde{T}}$ using (bind). We conclude $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} T(A_{\widetilde{T}} \Rightarrow B_{\widetilde{T}})$ using (factorization).
- <u>Imp-E</u>: Suppose that $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} T(A_{\widetilde{T}} \Rightarrow B_{\widetilde{T}})$ and $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} TA_{\widetilde{T}}$. We get $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} TA_{\widetilde{T}} \Rightarrow TB_{\widetilde{T}}$ using Proposition 2(4). We conclude $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} TB_{\widetilde{T}}$ using Imp-E.

- T-Elim: We directly have $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} TTA_{\widetilde{T}} \Rightarrow TA_{\widetilde{T}}$ using Proposition 2(1). We derive $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} T(TA_{\widetilde{T}} \Rightarrow A_{\widetilde{T}})$ using (factorization). Since we know $\vdash_{\mathsf{L}} (TA)_{\widetilde{T}} \Leftrightarrow TA_{\widetilde{T}}$, we conclude $\Gamma_{\widetilde{T}} \vdash_{\mathsf{L}} T(TA \Rightarrow A)_{\widetilde{T}}$.

For the second item, we show $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} t_{\widetilde{T}} = t$ by adapting the proof of Lemma 10. It follows that $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} A_{\widetilde{T}} \Leftrightarrow A$. We conclude $\vdash^{\mathfrak{efp}}_{\mathsf{L+T}} A^{\widetilde{T}} \Leftrightarrow A$ using (unit) and T-Elim. ◀

In the presence of equality, the soundness property for the factorizable monad translation is simplified compared to the one for the monad translation. We can drop the formula $\Delta^{\mathfrak{ep}}$ of Theorem 11, as the PropExt case can now be proved without any additional hypothesis.

▶ **Theorem 20** (Soundness for equality). *Let T be a factorizable monad operator of L such that $\vdash_{\mathsf{L}} (TA)_{\widetilde{T}} \Leftrightarrow TA_{\widetilde{T}}$ for any formula A. Let Γ, A be higher-order formulas.*

1. *For $* \in \{\mathfrak{e}, \mathfrak{ep}\}$, if $\Gamma \vdash^{*}_{\mathsf{L+T}} A$ then $\Gamma_{\widetilde{T}} \vdash^{*}_{\mathsf{L}} A^{\widetilde{T}}$.*
2. *For $* \in \{\mathfrak{ef}, \mathfrak{efp}\}$, if $\Gamma \vdash^{*}_{\mathsf{L+T}} A$ then $\Delta^{\mathfrak{ef}}, \Gamma_{\widetilde{T}} \vdash^{*}_{\mathsf{L}} A^{\widetilde{T}}$.*

Proof. For the first item, we complete the proof of Theorem 19 with the following cases:

- Eq-I and Eq-E: The proofs are the same as for Theorem 11.
- PropExt: Suppose $\Gamma_{\widetilde{T}} \vdash^{*}_{\mathsf{L}} T(A_{\widetilde{T}} \Rightarrow B_{\widetilde{T}})$ and $\Gamma_{\widetilde{T}} \vdash^{*}_{\mathsf{L}} T(B_{\widetilde{T}} \Rightarrow A_{\widetilde{T}})$. Using PropExt we have $\vdash^{*}_{\mathsf{L}} (A_{\widetilde{T}} \Rightarrow B_{\widetilde{T}}) \Rightarrow (B_{\widetilde{T}} \Rightarrow A_{\widetilde{T}}) \Rightarrow (A_{\widetilde{T}} = B_{\widetilde{T}})$, and we get $\vdash^{*}_{\mathsf{L}} T(A_{\widetilde{T}} \Rightarrow B_{\widetilde{T}}) \Rightarrow T(B_{\widetilde{T}} \Rightarrow A_{\widetilde{T}}) \Rightarrow T(A_{\widetilde{T}} = B_{\widetilde{T}})$ using Proposition 2(2) and Proposition 2(4). We conclude $\Gamma_{\widetilde{T}} \vdash^{*}_{\mathsf{L}} T(A_{\widetilde{T}} = B_{\widetilde{T}})$.

For the second item, we reuse most cases of the first item using weakening on $\Delta^{\mathfrak{ef}}$. The case FunExt corresponds to the one of Theorem 11, with $\Delta^{\mathfrak{ef}}$ instead of Δ^{*}. ◀

Unlike the monad translation of Section 3, the factorizable monad translation allows us to fully recover the specific behavior of the extension of Kuroda's double-negation translation to higher-order logic [21]. Specifically, monad operators are not inserted after implications, and the extra assumption $\Delta^{\mathfrak{ep}}$ is not needed for the soundness property in the presence of propositional equality.

7.3 Additional Embeddings between Logics

The embeddings of classical logic into intuitionistic logic of Corollary 12 are also valid with the factorizable monad translation. These embeddings introduce fewer monad operators than the monad translation.

▶ **Corollary 21** (Embeddings of classical logic into intuitionistic logic). *Let Γ, A be higher-order formulas.*

1. *If $\Gamma \vdash_{\mathsf{CL}} A$ then $\Gamma_{\widetilde{T}} \vdash_{\mathsf{IL}} A^{\widetilde{T}}$ with $TA := (A \Rightarrow \bot) \Rightarrow \bot$.*
2. *If $\Gamma \vdash_{\mathsf{CL}} A$ then $\Gamma_{\widetilde{T}} \vdash_{\mathsf{IL}} A^{\widetilde{T}}$ with $TA := (A \Rightarrow \bot) \Rightarrow A$.*
3. *If $\Gamma \vdash_{\mathsf{CL}} A$ then there exists a formula R such that $\Gamma_{\widetilde{T_R}} \vdash_{\mathsf{IL}} A^{\widetilde{T_R}}$ with $T_R A := (A \Rightarrow R) \Rightarrow A$.*

Moreover, for each of the above, we have $\vdash^{\mathfrak{efp}}_{\mathsf{CL}} A^{\widetilde{T}} \Leftrightarrow A$.

Proof. The proof is the one of Corollary 12, but using Theorem 19 instead of Theorem 8. ◀

▶ Remark 22. Kuroda's translation embeds classical logic into intuitionistic logic. It can also embed classical logic into minimal logic when the classical inference rule considered is either PEM or DNE1.

$$\frac{}{\Gamma \vdash A \vee \neg A}\ \text{PEM} \qquad \frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}\ \text{DNE1} \qquad \frac{}{\Gamma \vdash \neg\neg A \Rightarrow A}\ \text{DNE2}$$

But Kuroda's translation does not embed classical logic into minimal logic when we consider DNE2. As noted by Ferreira and Oliva [8], the modified Kuroda's translation, in which double negations are also inserted after implications, does embed classical logic into minimal logic, regardless of the classical inference rule considered. This is because the double-negation operator $TA := (A \Rightarrow \bot) \Rightarrow \bot$ satisfies the (factorization) property in intuitionistic logic, but not in minimal logic.

8 Conclusion and Future Work

We defined two monad embeddings for higher-order logic: the monad translation and the factorizable monad translation. The former broadens the scope of Kuroda's translation to any monad operator, but it additionally introduces monad operators after implications as a trade-off. The latter directly generalizes Kuroda's translation to factorizable monad operators. For both translations, the soundness property holds. As we are in higher-order logic, the characterization property holds when we assume both functional extensionality and propositional extensionality. We gave conditions so that the soundness property holds in the presence of functional extensionality and propositional extensionality.

These results generalize to monad operators the approach taken for extending Kuroda's translation to higher-order logic [21], and they extend to higher-order logic the approach taken for generalizing the double-negation translations to monad operators [1, 6, 22]. The monad translation and the factorizable monad translation entail various embeddings–depending on the monad operator and on the translation–of higher-order classical logic into higher-order intuitionistic logic, and of higher-order intuitionistic logic into higher-order minimal logic.

For higher-order coherent formulas, we even showed that intuitionistic provability entails minimal provability, and that classical provability entails intuitionistic provability. Coherent formulas therefore correspond to a constructive fragment of higher-order classical logic: for this fragment, we have provided an algorithm that transforms any classical proof into an intuitionistic proof.

Although originating from topos theory, Barr's theorem admits proof-theoretical demonstrations for the first-order finitary case [18, 16] and the first-order infinitary case [20, 17]. We gave a proof of the higher-order finitary case, by extending Palmgren's idea [18] of using Friedman's translation. The higher-order infinitary case remains to be investigated. In particular, simple type theory–used to model higher-order logic–is not sufficient to model an infinitary disjunction.

References

1. Peter Aczel. The Russell–Prawitz modality. *Mathematical. Structures in Comp. Sci.*, 11(4):541–554, August 2001. doi:10.1017/S0960129501003309.
2. Michael Barr. Toposes without points. *Journal of Pure and Applied Algebra*, 5(3):265–280, 1974. doi:10.1016/0022-4049(74)90037-1.
3. Chad E. Brown and Christine Rizkallah. Glivenko and Kuroda for simple type theory. *The Journal of Symbolic Logic*, 79(2):485–495, 2014. doi:10.1017/JSL.2013.10.
4. Alonzo Church. A Formulation of the Simple Theory of Types. *The Journal of Symbolic Logic*, 5(2):56–68, 1940. doi:10.2307/2266170.
5. Almudena Colacito, Dick de Jongh, and Ana Lucia Vargas. Subminimal negation. *Soft Computing*, 21:165–174, 2016. doi:10.1007/s00500-016-2391-8.
6. Martín Escardó and Paulo Oliva. The Peirce translation. *Annals of Pure and Applied Logic*, 163(6):681–692, 2012. Computability in Europe 2010. doi:10.1016/j.apal.2011.11.002.

7 Matt Fairtlough and Michael Mendler. Propositional Lax Logic. *Information and Computation*, 137:1–33, 1997. doi:10.1006/inco.1997.2627.

8 Gilda Ferreira and Paulo Oliva. On the Relation Between Various Negative Translations. In Ulrich Berger, Hannes Diener, Peter Schuster, and Monika Seisenberger, editors, *Logic, Construction, Computation*, pages 227–258. De Gruyter, Berlin, Boston, 2012. doi:10.1515/9783110324921.227.

9 Harvey Friedman. Classically and intuitionistically provably recursive functions. In Gert H. Müller and Dana S. Scott, editors, *Higher Set Theory*, pages 21–27, Berlin, Heidelberg, 1978. Springer Berlin Heidelberg.

10 Gerhard Gentzen. Die Widerspruchsfreiheit der Reinen Zahlentheorie. *Mathematische Annalen*, 112:493–565, 1936.

11 Valery Glivenko. Sur quelques points de la logique de M. Brouwer. *Bulletins de la classe des sciences*, 15:183–188, 1928.

12 Kurt Gödel. Zur intuitionistischen Arithmetik und Zahlentheorie. *Ergebnisse eines Mathematischen Kolloquiums*, 4:34–38, 1933.

13 Andrey Nikolaevich Kolmogorov. O principe tertium non datur. *Matematicheskiĭ Sbornik*, 32:646–667, 1925.

14 Sigekatu Kuroda. Intuitionistische Untersuchungen der formalistischen Logik. *Nagoya Mathematical Journal*, 2:35–47, 1951. doi:10.1017/S0027763000010023.

15 Eugenio Moggi. Notions of computation and monads. *Information and Computation*, 93(1):55–92, 1991. Selections from 1989 IEEE Symposium on Logic in Computer Science. doi:10.1016/0890-5401(91)90052-4.

16 Sara Negri. Contraction-free sequent calculi for geometric theories with an application to Barr's theorem. *Archive for Mathematical Logic*, 42:389–401, May 2003. doi:10.1007/s001530100124.

17 Sara Negri. Geometric rules in infinitary logic. In Ofer Arieli and Anna Zamansky, editors, *Arnon Avron on Semantics and Proof Theory of Non-Classical Logics*, pages 265–293. Springer International Publishing, Cham, 2021. doi:10.1007/978-3-030-71258-7_12.

18 Erik Palmgren. An Intuitionistic Axiomatisation of Real Closed Fields. *Mathematical Logic Quarterly*, 48(2):297–299, 2002. doi:10.1002/1521-3870(200202)48:2\%3C297::AID-MALQ297\%3E3.0.CO;2-G.

19 Luiz Carlos Pereira and Edward Hermann Haeusler. On constructive fragments of classical logic. In Heinrich Wansing, editor, *Dag Prawitz on Proofs and Meaning*, pages 281–292. Springer International Publishing, Cham, 2015. doi:10.1007/978-3-319-11041-7_12.

20 Michael Rathjen. Remarks on Barr's Theorem: Proofs in Geometric Theories. In Dieter Probst and Peter Schuster, editors, *Concepts of Proof in Mathematics, Philosophy, and Computer Science*, pages 347–374. De Gruyter, Berlin, Boston, 2016. doi:10.1515/9781501502620-019.

21 Thomas Traversié. Kuroda's Translation for Higher-Order Logic. Submitted, April 2024. doi:10.48550/arXiv.2404.19503.

22 Benno van den Berg. A Kuroda-style j-translation. *Archive for Mathematical Logic*, 58(5-6):627–634, August 2019. doi:10.1007/s00153-018-0656-x.