# Computational logic based on linear logic and fixed points

Matteo Manighetti, Dale Miller

# Computational logic based on linear logic and fixed points

Matteo Manighetti and Dale Miller

Inria Saclay & LIX, Ecole Polytechnique, France

**Abstract.** We use $\mu$MALL, the logic that results from adding least and greatest fixed points to first-order multiplicative-additive linear logic, as a framework for presenting several topics in computational logic. In particular, we present various levels of restrictions on the roles of fixed points in proofs and show that these levels capture different topics. For example, level 0 of $\mu$MALL captures (generalized) unification problems, level 1 captures Horn-clause logic programming, level 2 captures various model checking problems, and level 3 introduces a linearized form of arithmetic. We also show how the proof search interpretation of $\mu$MALL can be used to compute general recursive functions. Finally, we identify several situations where provability in Peano Arithmetic can be replaced by provability in $\mu$MALL. In such situations, the proof theory of $\mu$MALL can be used to study and implement proof search procedures for fragments of Peano Arithmetic.

## 1 Introduction

A feature of first-order logic is not only the presence of propositional connectives, first-order quantifiers, and first-order terms but also the class of non-logical constants usually called *predicates* that stand for relations between terms. When we move from first-order logic to first-order arithmetic, we introduce induction principles and banish undefined predicates by formally defining relations between terms. When moving from classical logic to arithmetic in this fashion, one arrives at a presentation of Peano Arithmetic. In this paper, we propose to study arithmetic based instead on linear logic by working directly with $\mu$MALL, which was first defined and studied in [3,4,6].

Linear logic has played various roles in computational logic. Many applications rely on the ability of linear logic to capture the multiset rewriting paradigm, which, in turn, allows a direct encoding of Petri nets [15], process calculi [24,29], and stateful computations [22,30]. Our use of linear logic here will have none of that flavor. While the sequent calculus we use to present $\mu$MALL in Section 3 is based on multisets of formulas, we shall not model computation as some rewriting of multisets of atomic-formulas-as-tokens. In contrast, when we use linear logic connectives within arithmetic, we capture computation and deduction via familiar means that rely on relations of numerical expressions. Our approach to linearized arithmetic is similar to that expressed recently by Girard [19] about linear logic: "*Linear* logic is an unfortunate expression that suggests a particular system, while it is the key permitting the abandonment of all systems."[1]

[1] The original French: "Logique *linéaire* est une expression malvenue qui suggère un système alors qu'il s'agit de la clef permettant de les abandonner tous."

Here, we propose linearized arithmetic not to have a new, non-standard arithmetic but to better understand computation and reasoning in arithmetic.

The contributions of this paper are the following.

1. We show that $\mu$MALL is a single *framework* for a range of common computational logic topics—such as unification, Horn-clause logic programming, model checking, and arithmetic—that have not been traditionally treated by one proof system.
2. We identify restrictions on the inference rules of $\mu$MALL as a way to classify and separate computational topics.
3. We use the proof theory of $\mu$MALL to describe how proof-search instead of proof-normalization can be used to compute functions from proofs that a relation is total and determinant.
4. Finally, we identify situations where provability in Peano Arithmetic can be replaced with provability in the weaker setting of $\mu$MALL.

There are several actual and potential advantages with using $\mu$MALL as a framework for computational logic. First, $\mu$MALL appears to be the only sequent calculus proof system for a significant part of arithmetic for which cut-elimination has been proved, and a *focused proof system* is known to be complete [4,6]. Focused proof systems have been used to design both *foundational proof certificates* [21] as well as large scale, synthetic inference rules with, for example, complex arithmetic side conditions. Second, the absence of contraction in a (cut-free) proof system (such as the proof system for $\mu$MALL) often provides the basis for designing complete proof-search algorithms: see, for example, [7,13]. Finally, all the connectives of $\mu$MALL (including the least and greatest fixed point operators) are *canonical* in the sense that two versions of a connective with the same rules are provably equivalent. Unfortunately, the operators of linear logic that provide for unbounded behaviors—the exponentials—are not canonical: as a result, many versions of exponentials can exist within linear logic (see, for example, [12,33]), which seems problematic when one is considering a formalization of Peano arithmetic.

## 2 Terms and formulas

We use Church's approach [9] to defining terms, formulas, and abstractions over these by making them all simply typed $\lambda$-terms. The primitive type $o$ denotes formulas (of linear and classical logics). For the scope of this paper, we assume that there is a second primitive type $\iota$ and that the (ambient) signature $\Sigma_0$ contains the constructors $z\colon \iota$ (zero) and $s\colon \iota \to \iota$ (successor). We abbreviate the terms $z$, $(s\ z)$, $(s\ (s\ z))$, $(s\ (s\ (s\ z)))$, etc by $\mathbf{0}$, $\mathbf{1}$, $\mathbf{2}$, $\mathbf{3}$, etc.

### 2.1 Logical connectives involving type $\iota$

We first present the logical connectives that relate to first-order structures. The two quantifiers $\forall$ and $\exists$ are both given the type $(\iota \to o) \to o$: the terms $\forall(\lambda x.B)$ and $\exists(\lambda x.B)$ of type $o$ are abbreviated as $\forall x.B$ and $\exists x.B$, respectively. Equality $=$ and non-equality $\neq$ are both of the type $\iota \to \iota \to o$. For $n \geq 0$, the least fixed point operator of arity $n$

is written as $\mu_n$ and the greatest fixed point operator of $n$-ary is written as $\nu_n$, and they both have the type $(A \to A) \to A$ where $A$ is the type $\iota \to \cdots \to \iota \to o$ in which there are $n$ occurrences of $\iota$. We seldom write explicitly the arity of fixed points since that can usually be determined from context when its value is important. The De Morgan dual of $\mu$ is $\nu$ and of $=$ is $\neq$.

Our formalizations of arithmetic do not contain predicate symbols: we do not admit any non-logical symbols of type $\iota \to \cdots \to \iota \to o$. As a result, there are no atomic formulas, usually defined as formulas with a non-logical symbol as their head. Equality, non-equality, and the fixed point operators are treated as logical connectives since they will all receive introduction rules in the sequent calculus proof systems we introduce soon.

### 2.2 Propositional connectives

The eight linear logic connectives for MALL are the following.

| | conjunction | true | disjunction | false |
|---|---|---|---|---|
| multiplicative | $\otimes$ | 1 | $\parr$ | $\bot$ |
| additive | & | $\top$ | $\oplus$ | 0 |

The four binary connectives have type $o \to o \to o$ and the four units have type $o$. (The use of 0 and 1 as logical connectives is unfortunate for a paper about arithmetic. We shall write numerals in boldface.) Formulas involving the set of logical connectives in Section 2.1 and these propositional connectives will be called $\mu$MALL formulas. This mixture of MALL connectives with the treatment of first-order structures given by first-order quantifiers, equality, and fixed points was presented in [6] and many of its proof-theoretic properties were established in [3,4].

Negation is not a logical connective: instead, when $B$ is a formula, we write $\overline{B}$ to denote the formula that results from taking the De Morgan dual of $B$. We occasionally use the linear implication $B \multimap C$ as an abbreviation for $\overline{B} \parr C$. We also use this overline notation when $B$ is the body of a fixed point expression, *i.e.*, when $B$ has the form $\lambda p \lambda \vec{x}.C$ where $C$ is a formula, $p$ is a first-order predicate variable, and $\vec{x}$ is a list of first-order variables, then $\overline{B}$ is $\lambda p \lambda \vec{x}.\overline{C}$ [4, Definition 2.1]. For example, if $B$ is $[\lambda p \lambda x.x = z \oplus \exists y.x = (s\ (s\ y)) \otimes p\ y]$ then $\overline{B}$ is $[\lambda p \lambda x.x \neq z\ \&\ \forall y.x \neq (s\ (s\ y)) \parr p\ y]$.

The connectives of linear logic are given a *polarity* as follows. The connectives $\parr$, $\bot$, &, $\top$, $\forall$, $\neq$, and $\nu$ are all *negative* while their De Morgan duals are all *positive*. A $\mu$MALL formula is positive or negative depending only on the polarity of its top-most connective. Note that the polarity flips between $B$ and $\overline{B}$. A formula is *purely positive* (resp., *purely negative*) if every logical connective it contains is positive (resp., negative).

## 3 Sequent calculus proof systems for $\mu$MALL and $\mu$LK

Provability in MALL is decidable. In a desire to move from MALL to a more general logic, Girard added to MALL the exponentials !, ? to yield full linear logic. While linear logic does not allow for the general application of the structural rules of weakening and

contraction, it does allow for these rules to be applied to certain instances of formulas labeled by these exponentials. By this controlled reinsertion of contraction, provability in linear logic becomes undecidable. The $\mu$MALL proof system takes a different approach to permitting unbounded behaviors to be encoded by fixed points.

The one-sided sequent calculus proof system for $\mu$MALL is composed of the following two sets of inference rules (taken from [6]). Figure 1 presents the rules that do not mention the least and greatest fixed points while the rules for the fixed points are in Figure 2. The variable $y$ in the $\forall$-introduction rule is an *eigenvariable*: it is restricted to not be free in any formula in the conclusion of that rule. The application of a substitution $\theta$ to a signature $\Sigma$ (written $\Sigma\theta$ in the $\neq$ rule in Figure 1) is the signature that results from removing from $\Sigma$ the variables in the domain of $\theta$ and adding back any variables that are free in the range of $\theta$. In the $\neq$-introduction rule, if the terms $t$ and $t'$ are not unifiable, then the premise is empty and the conclusion is immediately proved.

If we were working in a two-sided sequent calculus, the $\nu$-rule in Figure 2 could be written in the following two ways.

$$\frac{\Gamma \vdash \Delta, S\,\vec{t} \quad S\,\vec{x} \vdash B S\,\vec{x}}{\Gamma \vdash \nu B \vec{t}, \Delta}\ \textit{coinduction} \qquad \frac{\Gamma, S\,\vec{t} \vdash \Delta \quad B S\,\vec{x} \vdash S\,\vec{x}}{\Gamma, \mu B \vec{t} \vdash \Delta}\ \textit{induction}$$

That is, the one rule for $\nu$ yields both coinduction and induction. In general, we shall speak of the higher-order substitution term $S$ used in both of these rules as the *invariant* of that rule (*i.e.*, we will not use the term co-invariant even though that might be more appropriate in some settings).

As we mentioned above, we use Church's *Simple Theory of Types* (STT) approach to representing formulas, terms, and abstractions over these [9]. This approach provides a great advantage for us here. Not only does STT treat first-order formulas naturally (by identifying the binding in quantifiers with $\lambda$-bindings), it also provides an elegant treatment of higher-order substitutions (needed for handing induction invariants), as well as provides a simple treatment of fixed point expressions and the binding mechanisms used there. In particular, we shall assume that formulas in sequents are always treated modulo $\alpha\beta\eta$-conversion. When presenting sequents, we usually display formulas in $\lambda$-normal form. Note that formula expressions such as $B\ S\ \vec{t}$ (see Figure 2) are parsed as $(((B\ S)t_1)t_2)$ if $\vec{t}$ is the list of terms $t_1, t_2$.

We make the following observations about this proof system.

1. The $\mu\nu$ rule is a limited form of the initial rule. The general form of the initial rule, namely, that the sequent $\vdash Q, \overline{Q}$ is provable, is admissible: this more general rule is named init.
2. The rule for $\mu$ allows for the $\mu$ fixed point to be unfolded. This rule captures, in part, the identification of $\mu B$ with $B(\mu B)$; that is, that $\mu B$ is a fixed point of $B$. This inference rules allows one occurrence of $B$ in $(\mu B)$ to be expanded to two occurrences of $B$ in $B(\mu B)$. In this way, unbounded behaviors can appear in $\mu$MALL where it did not occur in MALL.
3. The *unfold* rule in Figure 3, which simply unfolds $\nu$-expression, is admissible in this proof system by using the $\nu$-rule with the invariant $S = B(\nu B)$.

*Unpolarized formulas* are built from the connectives presented in Section 2.1 along with the connectives $\wedge$, $tt$, $\vee$, $ff$ (instead of the MALL connectives given in Section 2.2).

$$\frac{\vdash \Gamma, P \quad \vdash \Delta, Q}{\vdash \Gamma, \Delta, P \otimes Q}\ \otimes \qquad \frac{}{\vdash 1}\ 1 \qquad \frac{\vdash \Gamma, P, Q}{\vdash \Gamma, P \parr Q}\ \parr \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \bot}\ \bot$$

$$\frac{\vdash \Gamma, P \quad \vdash \Gamma, Q}{\vdash \Gamma, P \,\&\, Q}\ \& \qquad \frac{}{\vdash \Delta, \top}\ \top \qquad \frac{\vdash \Gamma, P_i}{\vdash \Gamma, P_0 \oplus P_1}\ \oplus$$

$$\frac{\{\ \vdash \Gamma\theta : \theta = mgu(t, t')\ \}}{\vdash \Gamma, t \neq t'}\ \neq \qquad \frac{}{\vdash t = t}\ = \qquad \frac{\vdash \Gamma, Pt}{\vdash \Gamma, \exists x.Px}\ \exists \qquad \frac{\vdash \Gamma, Py}{\vdash \Gamma, \forall x.Px}\ \forall$$

**Fig. 1.** The core set of inference rules

$$\frac{\vdash \Gamma, S\vec{t} \quad \vdash BS\vec{x}, \overline{(S\vec{x})}}{\vdash \Gamma, \nu B\vec{t}}\ \nu \qquad \frac{\vdash \Gamma, B(\mu B)\vec{t}}{\vdash \Gamma, \mu B\vec{t}}\ \mu \qquad \frac{}{\vdash \mu B\vec{t}, \nu\overline{B}\vec{t}}\ \mu\nu$$

**Fig. 2.** Add these rules to the core set to get $\mu$MALL

$$\frac{\vdash \Gamma, B(\nu B)\vec{t}}{\vdash \Gamma, \nu B\vec{t}}\ \mathit{unfold} \qquad \frac{\vdash \Gamma, Q, Q}{\vdash \Gamma, Q}\ C \qquad \frac{\vdash \Gamma}{\vdash \Gamma, Q}\ W \qquad \frac{\vdash \Gamma, Q \quad \vdash \Delta, \overline{Q}}{\vdash \Gamma, \Delta}\ \mathit{cut}$$

**Fig. 3.** Some additional rules

For convenience, we will occasionally allow implications in classical formulas: in those cases, we treat $P \supset Q$ as $\overline{P} \vee Q$ where $\overline{P}$ is the negation normal form of the negation of $P$. A $\mu$MALL formula $\hat{Q}$ is a *polarized* version of the classical formula $Q$ if every occurrence of & and $\otimes$ in $\hat{Q}$ is replaced by $\wedge$ in $Q$, every occurrence of $\parr$ and $\oplus$ in $\hat{Q}$ is replaced by $\vee$ in $Q$, every occurrence of 1 and $\top$ in $\hat{Q}$ is replaced by *tt* in $Q$, and every occurrence of 0 and $\bot$ in $\hat{Q}$ is replaced by *ff* in $Q$. Notice that if $Q$ has $n$-occurrences of propositional connectives, then there are $2^n$ formulas $\hat{Q}$ that are polarized versions of $Q$. We never mix polarized and unpolarized connectives in the same formulas. When we think to formulas of arithmetic, we usually identify them with unpolarized formulas.

We define the following four *levels of restriction* on proof systems that involve the fixed point operators $\mu$ and $\nu$. These restrictions are based solely on occurrences of the $\mu\nu$ inference rules and the introduction rules for $\mu$ and $\nu$.

**Level 0** Use only the rules in Figure 1.

**Level 1** Add to Level 0 the $\mu$ rule and the $\nu$-unfolding rule *unfold*. Here, both fixed points can only be unfolded.

**Level 2** Add to Level 1 the $\nu$ inference rules. The *unfold* rule is redundant here. This level, therefore, contains only the $\nu$ and $\mu$ rules from Figure 2.

**Level 3** Add to Level 2 the rule $\mu\nu$. This is $\mu$MALL.

We write $\mu\text{MALL}^l$ to denote the proof systems that results from imposing the level $l \in \{0, 1, 2\}$ restriction. In subsequent sections, we shall see how certain activities in computational logic use $\mu$MALL proofs that satisfy different level restrictions.

We define $\mu$LK to be the proof system $\mu$MALL but with the inference rules for contraction $C$, weakening $W$, and cut (all three from Figure 3) added to $\mu$MALL. The $\mu$LK proof system, like the $\mu$MALL proof system, uses only polarized formulas. For

$l \in \{0, 1, 2\}$, the proof system $\mu\text{LK}^l$ is $\mu\text{MALL}^l$ with contraction, weakening, and cut added.

*Example 1.* The formula $\forall x \forall y[x = y \vee x \neq y]$ can be polarized as either

$$\forall x \forall y[x = y \parr x \neq y] \quad \text{or} \quad \forall x \forall y[x = y \oplus x \neq y].$$

Only the first of these two polarized formulas is provable in $\mu$MALL, although both formulas are provable in $\mu$LK (in fact, in $\mu\text{LK}^0$).

Whether or not cut can be eliminated from $\mu$LK is currently open. We can prove, however, that $\mu$LK is consistent (see the following theorem) and that Peano Arithmetic is included within $\mu$LK (Theorem 4).

**Theorem 1.** *$\mu$LK is consistent.*

The proof of this theorem can be found in the Appendix A. It is worth noting that adding contraction to some logical systems with weak forms of fixed points can change that logic from being consistent to inconsistent. For example, both Girard [18] and Schroeder-Heister [35] describe a linear logic with unfolding fixed points that is consistent, but when contraction is added, the logic becomes inconsistent. In their case, negations are allowed in the body of fixed point definitions. The theorem above proves that adding contraction to $\mu$MALL does not lead to inconsistency.

## 4 Generalized unification problems

The restriction level 0 yields the proof system $\mu\text{MALL}^0$ for formulas that do not contain fixed point operators. Such formulas are quantified MALL formulas only over the judgments for equality and non-equality and they naturally capture various generalizations of unification problems. More specifically, the first-order unification problem containing the free variables $x_1, \ldots, x_n$ and the disagreement pairs $\langle t_1, t'_1 \rangle, \ldots, \langle t_m, t'_m \rangle$ can be encoded as the quantified formula

$$\exists x_1 \ldots \exists x_n \, [t_1 = t'_1 \wedge \ldots \wedge t_m = t'_m] \qquad (n, m \geq 0).$$

(Higher-order unification problems have also been encoded in this fashion [2].) When implementing proof search in proof systems containing eigenvariables or term-level bindings [32,34], such unification problems are often generalized to contain mixed quantification [28]: that is, they are encoded as formulas of the form

$$Q_1 x_1 \ldots Q_n x_n \, [t_1 = t'_1 \wedge \ldots \wedge t_m = t'_m] \qquad (n, m \geq 0),$$

where $Q_i$ is either $\forall$ or $\exists$. A more expressive extension, however, would allow replacing simple equations with implicational judgments among equations. Consider the following two classes of formulas.

$$\begin{aligned} \Phi &::= \Phi \wedge \Phi \mid \exists x.\Phi \mid \forall x.\Phi \mid \Psi \\ \Psi &::= t_1 = t'_1 \supset \cdots \supset t_n = t'_n \supset t_0 = t'_0 \quad (n \geq 0) \end{aligned}$$

We call formulas of the form $\Phi$ *generalized unification problems*. The non-equality of two terms $t$ and $t'$ can be encoded as the formula $t = t' \supset c = d$ where $c$ and $d$ are two distinct closed terms. These generalized unification problems extend the equational problems studied in [11] in which quantifier prefixes are restricted to be of the form $\exists\forall$.

*Example 2.* It is easy to prove that equality is an equivalence relation. In particular, the formula $\forall x\forall y\forall w.(x \neq y \parr y \neq w \parr x = w)$ is a polarized form of the axiom of transitivity for equality. The following is a proof of this formula.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash x = x}\,=}{\vdash x \neq w, x = w}\,\neq,\ \text{replace } w \text{ with } x}{\vdash x \neq y, y \neq w, x = w}\,\neq,\ \text{replace } y \text{ with } x}{\vdash x \neq y \parr y \neq w \parr x = w}\,\parr \times 2}{\vdash \forall x\forall y\forall w.(x \neq y \parr y \neq w \parr x = w)}\,\forall \times\ 3$$

Two other Peano axioms can also be proved directly in this logic. The two formulas

$$\forall x\forall y\,[(s\ x) \neq (s\ y) \parr x = y] \quad \text{and} \quad \forall x\,[(s\ x) \neq \mathbf{0}]$$

are polarized versions of the axioms that assert that successor is injective and that zero is not a successor. They both have simple $\mu$MALL proofs.

Consider the polarization scheme that replaces $\wedge$ with $\otimes$ and $t_1 = t'_1 \supset \cdots \supset t_n = t'_n \supset t_0 = t'_0$ with $t_1 \neq t'_1 \parr \cdots \parr t_n \neq t'_n \parr t_0 = t'_0$.

**Theorem 2.** *Let $\Phi$ be a generalized unification problem and let $\hat{\Phi}$ be the polarization of $\Phi$ following the scheme described above. Then, $\hat{\Phi}$ is provable in $\mu$LK if and only if $\hat{\Phi}$ is provable in $\mu$MALL.*

*Proof.* Since the rules in $\mu$MALL are all admissible in $\mu$LK, the converse implication is immediate. To prove the forward direction, we prove the more general statement: If $\vdash \Gamma$ is provable in $\mu$LK, where $\Gamma$ is a multiset of polarized versions of $\Phi$-formulas, then there is a formula $G \in \Gamma$ such that $\vdash G$ is provable in $\mu$MALL. We proceed by induction on the structure of such a $\mu$LK proof. If the last inference rule is either contraction or weakening, the result is immediate. If the last inference rule is cut, then there must be a side formula in one of the premises that is provable since it cannot be the case that both $Q$ and $\overline{Q}$ are provable (follows from the cut-elimination of $\mu$LK). All other cases for the inference rules are immediate except for the $\parr$ introduction, in which case it is not true in general. Fortunately, the general form of $\parr$ does not appear here. Instead, the $\parr$ rule is only applied with formulas of the form $t_1 \neq t'_1 \parr \cdots \parr t_n \neq t'_n \parr t_0 = t'_0$. Since the introduction rules for $\parr$ and $\neq$ are invertible, the proof follows immediately.

Thus, we have the result that provability in the classical and linear settings coincide: this means that contraction and weakening plays no role in such generalized unification problems. Note also that the additive connectives of $\mu$MALL are not needed.

If we consider only unification problems without implications (that is, force $n = 0$ in the description of $\Psi$ above), then provability of $\Phi$ formulas is decidable. However, when implications are permitted, provability of $\Phi$ formulas is undecidable [31].

## 5 Logic programming and model checking

The restriction level 1 yields the proof system $\mu\text{MALL}^1$ in which it is possible to unfold both $\mu$ and $\nu$ fixed points but no induction or coinduction rules are permitted in such proofs. As we show now, the proof theory of logic programming with Horn clauses is completely described by using $\mu\text{MALL}^1$ proofs of purely positive formulas.

The connection between Horn clauses and least fixed points is well-known and goes back to at least the *Clark completion* of Horn clauses [10]. To illustrate, consider the following two Horn clauses axiomatizing addition, written using Prolog-style syntax (meaning that :- denotes implied-by $\subset$).

$$\begin{array}{c} \forall N. \qquad \mathit{plus}\ \mathbf{0}\ N\ N. \\ \forall N \forall M \forall P.\ \mathit{plus}\ (s\ N)\ M\ (s\ P)\ \texttt{:-}\ \mathit{plus}\ N\ M\ P \end{array}$$

By moving the term structures in the head so they appear instead in the body of these clauses, we have the equivalent clauses

$$\begin{array}{l} \forall N \forall M \forall P.\ \mathit{plus}\ N\ M\ P\ \texttt{:-}\ N = \mathbf{0} \wedge M = P. \\ \forall N \forall M \forall P.\ \mathit{plus}\ N\ M\ P\ \texttt{:-}\ \exists N' \exists P'\ (N = (s\ N') \wedge P = (s\ P') \wedge \mathit{plus}\ N'\ M\ P'). \end{array}$$

These two clauses can now be merged into one by introducing a disjunction.

$$\begin{array}{l} \forall N \forall M \forall P.\ \mathit{plus}\ N\ M\ P\ \texttt{:-}\ (N = \mathbf{0} \wedge M = P) \vee \\ \qquad\qquad\qquad\qquad \exists N' \exists P'\ (N = (s\ N') \wedge P = (s\ P') \wedge \mathit{plus}\ N\ M\ P). \end{array}$$

This Prolog clause suggests using the following fixed point expression to define *plus*.

$$\mathit{plus} = \mu \lambda P \lambda n \lambda m \lambda p((n = \mathbf{0} \otimes m = p) \oplus \exists n' \exists p' (n = (s\ n') \otimes p = (s\ p') \otimes P\ n'\ m\ p'))$$

In general, all Horn clauses can be rewritten in this fashion so that all the predicates they define can be expressed as purely positive expressions.

An immediate consequence of Theorem 5 (Section 8) is the fact that if such a purely positive expression has a $\mu$LK proof, then it is also provable in $\mu$MALL. It is also clear that if there exists a $\mu$MALL proof of a purely positive formula, then that proof is actually a $\mu\text{MALL}^1$ proof. Finally, given that Horn clauses can interpret Turing machines [36], it is undecidable whether or not a purely positive expression has a $\mu$MALL proof.

*Example 3 (Taken from [21]).* Let the sets $A = \{\mathbf{0}, \mathbf{1}\}$ and $B = \{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$ be encoded as the $\lambda$-expressions $\lambda x.\ x = \mathbf{0} \vee x = \mathbf{1}$ and $\lambda x.\ x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}$, respectively. A polarized version of the formula $\forall x. Ax \supset Bx$ is $\forall x.[(x \neq \mathbf{0}\ \&\ x \neq \mathbf{1})\ \parr\ (x = \mathbf{0} \oplus x = \mathbf{1} \oplus x = \mathbf{2})]$ and this formula has the following $\mu$MALL proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \mathbf{0} = \mathbf{0}}\,=}{\vdash \mathbf{0} = \mathbf{0} \oplus \mathbf{0} = \mathbf{1} \oplus \mathbf{0} = \mathbf{2}}\,\oplus}{\vdash x \neq \mathbf{0}, x = \mathbf{0} \oplus x = \mathbf{1} \oplus x = \mathbf{2}}\,\neq \qquad \dfrac{\dfrac{\dfrac{}{\vdash \mathbf{1} = \mathbf{1}}\,=}{\vdash \mathbf{1} = \mathbf{0} \oplus \mathbf{1} = \mathbf{1} \oplus \mathbf{1} = \mathbf{2}}\,\oplus}{\vdash x \neq \mathbf{1}, x = \mathbf{0} \oplus x = \mathbf{1} \oplus x = \mathbf{2}}\,\neq}{\vdash x \neq \mathbf{0}\ \&\ x \neq \mathbf{1}, x = \mathbf{0} \oplus x = \mathbf{1} \oplus x = \mathbf{2}}\,\&}{\vdash \forall x.[(x \neq \mathbf{0}\ \&\ x \neq \mathbf{1})\ \parr\ (x = \mathbf{0} \oplus x = \mathbf{1} \oplus x = \mathbf{2})]}\,\forall, \parr$$

Here, the doubled horizontal line indicates that more than one inference rule is applied.

Switching to model checking terminology, proofs involving logic programs express *reachability* problems. But what about *non-reachability*? As has been shown [20], the notion of *negation as finite failure* can be captured (in our setting) by building a $\mu$MALL$^1$ proof of a purely negative expression. In this setting, the restriction to level 1 proofs is significant since that restriction rules out the use of invariants. Along a similar vein, the model checking problem of determining simulation and bisimulation of two transition systems is easily written as a negative formula with at most one alternation of polarities [21,27] (assuming that the transition systems are defined using purely positive expressions).

In general, capturing simulation with proofs restricted to level 1 requires that transition systems are acyclic. Capturing both non-reachability and simulation (and non-simulation) for cyclic transitions is possible by moving to restriction level 2. For example, consider a graph that may contain cycles and consider a proof that there is *no* path from, say, node $a$ to node $b$. This is provable by using an invariant $S$ that encodes a connected component containing $a$ but not $b$. The coinductive proof that one must then build must show that the set $S$ is closed under one-step transitions and contains $a$ and does not contain $b$. Such reasoning does not, surprisingly, need to use the $\mu\nu$ rule [21].

## 6 Function computation via proof search

Consider a binary relation on natural numbers $\phi(x, y)$. We say that $\phi$ encodes a function $f$ from $\mathbb{N}$ to $\mathbb{N}$ if $\phi(x, y)$ holds exactly when $f(x) = y$. The Curry-Howard approach to relating proof theory to computation [23] extracts from a natural deduction proof of $\forall x \exists y.\phi(x, y)$ a $\lambda$-term which can be seen as an algorithm for computing $f$. The algorithmic content of such a $\lambda$-term arises from a non-deterministic rewriting process that selects $\beta$-redexes for reduction. In most typed $\lambda$-calculus systems, any sequence of rewritings will end in the same normal form (strong normalization). Of course, some sequences of rewrites might be very long and others could be very short.

We now describe a different mechanism for computing the function that underlies a relation for which totality has been proved. This mechanism will not rely on the Curry-Howard correspondence nor on $\lambda$-terms. Instead, we rely directly on the mechanisms underlying proof search.

One way to prove that a binary relation $\phi$ encodes a function is to prove the *totality* and *determinancy* properties of $\phi$: that is, prove

$$[\forall x \exists y.\phi(x, y)] \wedge [\forall x \forall y_1 \forall y_2.\phi(x, y_1) \supset \phi(x, y_2) \supset y_1 = y_2].$$

Clearly, these properties imply that for every natural number $x$, the predicate $\lambda y.\phi(x, y)$ denotes a singleton set.

Note that if $P$ and $Q$ are predicates of arity one and if $P$ denotes a singleton, then $\exists x[Px \wedge Qx]$ and $\forall x[Px \supset Qx]$ are logically equivalent. We assume here that $Px$ is a purely positive expression with $x$ as its only free variable: as we have seen in Section 5, such expressions can capture general recursive predicates. Notice that the proof search semantics of these equivalent formulas are surprisingly different. In particular, if we attempt to prove $\exists x[Px \wedge Qx]$, then we must *guess* a term $t$ and then *check* that $t$ denotes the element of the singleton (by proving $P(t)$). In contrast, if we attempt to

prove $\forall x[Px \supset Qx]$ then we simply allocate an eigenvariable $y$ (which we will eventually instantiate with $t$) and then attempt to prove the sequent $\vdash Py \supset Qy$. Obviously, such an attempt at building a proof might actually *compute* the value $t$ (especially if we can restrict proofs to Level 1). Thus, singletons introduce an ambiguity of polarity: the positive formula $\exists x[Px \wedge Qx]$ can be changed to the negative formula $\forall x[Px \supset Qx]$ and vice versa.

*Example 4.* Consider the fixed point expression used for identifying natural numbers

$$nat = \mu\lambda N\lambda n(n = \mathbf{0} \oplus \exists n'(n = (s\ n') \otimes N\ n'))$$

and recall the least fixed point expression for *plus* given in Section 5. The Peano arithmetic formula $\forall n\ [nat\ n \supset plus\ n\ \mathbf{0}\ n]$ can be polarized as the $\mu$MALL formula $\forall n\ [\overline{(nat\ n)} \parr plus\ n\ \mathbf{0}\ n]$ and this formula has the following $\mu$MALL proof.

$$\dfrac{\dfrac{\dfrac{}{\vdash \overline{plus\ n\ \mathbf{0}\ n}, plus\ n\ \mathbf{0}\ n}\ \mu\nu \qquad \dfrac{\Xi_1 \qquad \Xi_2}{\vdash (n \neq \mathbf{0}\ \&\ \forall n'(n \neq (s\ n') \parr \overline{plus\ n'\ \mathbf{0}\ n'})),\ plus\ n\ \mathbf{0}\ n}\ \&, \forall, \parr, \neq}{\vdash \overline{(nat\ n)},\ plus\ n\ \mathbf{0}\ n}\ \nu}{\vdash \forall n\ [\overline{(nat\ n)} \parr plus\ n\ \mathbf{0}\ n]}\ \forall, \parr$$

$$\Xi_1 = \vdash plus\ \mathbf{0}\ \mathbf{0}\ \mathbf{0} \qquad \Xi_2 = \dfrac{}{\vdash \overline{plus\ n'\ \mathbf{0}\ n'},\ plus\ (s\ n')\ \mathbf{0}\ (s\ n')}$$

Here, the $\Xi_1$ subproof involves only introduction rules while the $\Xi_2$ subproof involves also an occurrence of the $\mu\nu$ rule.

*Example 5.* The following derivation can be seen as a partial computation of 2 + 2:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \mathbf{2} = (s\ \mathbf{1})}\ = \quad \dfrac{}{\vdash \mathbf{4} = (s\ \mathbf{3})}\ = \quad \vdash plus\ \mathbf{1}\ \mathbf{2}\ \mathbf{3}}{\vdash \mathbf{2} = (s\ \mathbf{1}) \otimes \mathbf{4} = (s\ \mathbf{3}) \otimes plus\ \mathbf{1}\ \mathbf{2}\ \mathbf{3}}\ \otimes \times 2}{\vdash \exists n' \exists p'(\mathbf{2} = (s\ n') \otimes \mathbf{4} = (s\ p') \otimes plus\ n'\ \mathbf{2}\ p')}\ \exists \times 2}{\vdash (\mathbf{2} = \mathbf{0} \otimes \mathbf{2} = \mathbf{4}) \oplus \exists n' \exists p'(\mathbf{2} = (s\ n') \otimes \mathbf{4} = (s\ p') \otimes P\ n'\ \mathbf{2}\ p')}\ \oplus}{\vdash plus\ \mathbf{2}\ \mathbf{2}\ \mathbf{4}}\ \mu}{\vdash \exists p.plus\ \mathbf{2}\ \mathbf{2}\ p}\ \exists$$

To complete this computation, we must construct a similar subproof verifying that 1 + 2 = 3. In particular, the witness used to instantiate the final $\exists p$ is, in fact, that sum. Unfortunately, proof construction in this system does not help us to construct the value of this sum. Instead, the first step in building such a proof bottom-up starts with guessing a value and then checking that it is the correct sum.

*Example 6.* Given the definition of addition on natural numbers above, it is an easy matter to prove in $\mu$MALL any polarized form of the following totality and determinancy theorems.

$$[\forall x_1 \forall x_2.\ nat\ x_1 \supset nat\ x_2 \supset \exists y.(plus(x_1, x_2, y) \wedge nat\ y)]$$
$$[\forall x_1 \forall x_2.\ nat\ x_1 \supset nat\ x_2 \supset \forall y_1 \forall y_2.\ plus(x_1, x_2, y_1) \supset plus(x_1, x_2, y_2) \supset y_1 = y_2]$$

These proofs require both induction and the $\mu\nu$ rule. Using the cut rule with (the obvious) proofs of *nat* **2** and *nat* **3**, we know that $\lambda y.(\mathit{plus}\ \mathbf{2}\ \mathbf{3}\ y)$ denotes a singleton. In order to compute the sole member of the singleton $\lambda y.(\mathit{plus}\ \mathbf{2}\ \mathbf{3}\ y)$, we could perform cut-elimination with the inductively proved totality theorem in this example. Instead of such a *proof-reduction* approach to computation, the *proof search* approach starts by replacing the goal $\exists y.(\mathit{plus}\ \mathbf{2}\ \mathbf{3}\ y \wedge \mathit{nat}\ y)$ with $\forall y.(\mathit{plus}\ \mathbf{2}\ \mathbf{3}\ y \supset \mathit{nat}\ y)$. Attempting to prove this second formula leads to an incremental construction of the answer substitution for $y$, namely, **5**.

Assume that $P$ is a purely positive formula and that we have a $\mu$MALL proof that $P$ is a singleton. As we stated above, this means that we have a $\mu$MALL proof of $\forall x[Px \supset \mathit{nat}\ x]$. This proof can be understood as a means to compute the unique element of $P$ except that there might be instances of the induction rule used to prove $\forall x[Px \supset \mathit{nat}\ x]$. If we can force, however, the proof of this latter formula to be restricted to level 1, then such a restricted proof provides an explicit computation. As the following example shows, it is not the case that if there is a $\mu$MALL proof of $\forall x[Px \supset \mathit{nat}\ x]$ then it also has a $\mu\mathrm{MALL}^1$ proof.

*Example 7.* Let $P$ be $\mu(\lambda R \lambda x.x = \mathbf{0} \oplus (R\ (s\ x)))$. Clearly, $P$ denotes the singleton set containing zero. There is also a $\mu$MALL proof that $\forall x[Px \supset \mathit{nat}\ x]$, but there is no (cut-free) $\mu\mathrm{MALL}^1$ proof of this theorem since just using unfoldings will lead to an unbounded proof search attempt which roughly follows the following outline.

$$\dfrac{\vdash \mathit{nat}\ \mathbf{0} \qquad \dfrac{\dfrac{\vdots}{\vdash \overline{P\ (s\ (s\ y))}, \mathit{nat}\ y}}{\vdash \overline{P\ (s\ y)}, \mathit{nat}\ y}\ \mathit{unfold}, \&, \neq}{\vdash \overline{P\ y}, \mathit{nat}\ y}\ \mathit{unfold}, \&, \neq$$

Although proof search can contain potentially unbounded branches, we can still use the proof search concepts of unification and non-deterministic search to compute the value within a singleton. We define a non-deterministic algorithm as follows. The *state* of this algorithm is a sequent-like structure

$$\langle x_1, \ldots, x_n\ ;\ B_1, \ldots, B_m\ ;\ \mathit{nat}\ t\rangle,$$

where $t$ is a term, $B_1, \ldots, B_m$ is a multiset of purely positive formulas, and all variables free in $t$ and in the formulas $B_1, \ldots, B_m$ are in the set of variables $x_1, \ldots, x_n$. A *success state* is one of the form $\langle \cdot\ ;\ \cdot\ ;\ \mathit{nat}\ t\rangle$ (that is, when $n = m = 0$) and that state is said to have *value* $t$.

Given the state $S = \langle \Sigma\ ;\ B_1, \ldots, B_m\ ;\ \mathit{nat}\ t\rangle$ with $m \geq 1$, we can non-deterministically select one of the $B_i$ formulas: for sake of simplicity, assume that we have selected $B_1$. We define the transition to another state, written as $S \Rightarrow S'$ depending on the top-level structure of $B_1$.

- If $B_1$ is $u = v$ and the terms $u$ and $v$ are unifiable with most general unifier $\theta$, then we transition to $\langle \Sigma\theta\ ;\ B_2\theta, \ldots, B_m\theta\ ;\ \mathit{nat}\ (t\theta)\rangle$.

– If $B_1$ is $B \otimes B'$ then we transition to $\langle \Sigma \,;\, B, B', B_2, \ldots, B_m \,;\, nat\ t\rangle$.
– If $B_1$ is $B \oplus B'$ then we transition to either

$$\langle \Sigma \,;\, B, B_2, \ldots, B_m \,;\, nat\ t\rangle \quad \text{or} \quad \langle \Sigma \,;\, B', B_2, \ldots, B_m \,;\, nat\ t\rangle.$$

– If $B_1$ is $\mu B \vec{t}$ then we transition to $\langle \Sigma \,;\, B(\mu B)\vec{t}, B_2, \ldots, B_m \,;\, nat\ t\rangle$.
– If $B_1$ is $\exists y.\ B\ y$ then we transition to $\langle \Sigma, y \,;\, B\ y, B_2, \ldots, B_m \,;\, nat\ t\rangle$ (assuming, of course, that $y$ is picked to not be in $\Sigma$).

This non-deterministic algorithm is essentially applying left-introduction rules in a bottom-up fashion and, in the event that there are two premises, selecting (non-deterministically) just one premise to follow. We abbreviate

$$(\exists x.P\ x) \wedge (\forall x_1 \forall x_2 . P\ x_1 \supset P\ x_2 \supset x_1 = x_2)$$

by $\exists! x.P\ x$ in the following theorem.

**Theorem 3.** *Assume that $P$ is a purely positive formula and that we have a $\mu$MALL proof of $\exists! y.Py \wedge \mathrm{nat}\ y$. Thus, $P$ denotes a singleton set and let $t$ be the element of that set. There is a sequence of transitions from the initial state $\langle y \,;\, P\ y \,;\, \mathrm{nat}\ y\rangle$ to a success state if and only if the value of that success state is $t$.*

*Proof.* We first define the notion of an *augmented state* as a structure of the form

$$\langle \Sigma \mid \theta \,;\, B_1 \mid \Xi_1, \ldots, B_m \mid \Xi_m \,;\, nat\ t\rangle,$$

where

– $\theta$ is a substitution with domain equal to $\Sigma$ and which has no free variables in its range, and
– for all $i \in \{1, \ldots, m\}$, $\Xi_i$ is a $\mu$MALL proof of $\theta(B_i)$.

Clearly, if we strike out the augmented items (in red), we are left with a regular state. Given that we have a $\mu$MALL proof of $\exists! y.Py$, we know that there must exist a $\mu$MALL proof $\Xi_0$ of $Pt$ for some term $t$. Note that there is no occurrence of induction in $\Xi_0$. The initial augmented state is $\langle y \mid [y \mapsto t] \,;\, Py \mid \Xi_0 \,;\, nat\ y\rangle$. As we detail now, the proof structures $\Xi_i$ provide oracles that steer this non-deterministic algorithm to a success state with value $t$. Given the augmented state $\langle \Sigma \mid \theta \,;\, B_1 \mid \Xi_1, \ldots, B_m \mid \Xi_m \,;\, nat\ s\rangle$, we consider selecting the first pair $B_1 \mid \Xi_1$ and consider the cases for $B_1$.

– If $B_1$ is $B' \otimes B''$ then the last inference rule of $\Xi_1$ is $\otimes$ with premises $\Xi'$ and $\Xi''$, and we make a transition to $\langle \Sigma \mid \theta \,;\, B' \mid \Xi', B'' \mid \Xi'', \ldots, B_m \mid \Xi_m \,;\, nat\ s\rangle$.
– If $B_1$ is $B' \oplus B''$ then the last inference rule of $\Xi_1$ is $\oplus$ and that rule selects either the first or the second disjunct. In either case, let $\Xi'$ be the proof of its premise. Depending on which of these disjuncts is selected, we make a transition to either

$$\langle \Sigma \mid \theta \,;\, B' \mid \Xi', B_2 \mid \Xi_2, \ldots, B_m \mid \Xi_m \,;\, nat\ s\rangle \text{ or}$$
$$\langle \Sigma \mid \theta \,;\, B'' \mid \Xi', B_2 \mid \Xi_2, \ldots, B_m \mid \Xi_m \,;\, nat\ s\rangle, \text{ respectively.}$$

– If $B_1$ is $\mu B\vec{t}$ then the last inference rule of $\Xi_1$ is $\mu$. Let $\Xi'$ be the proof of the premise of that inference rule. We make a transition to

$$\langle \Sigma \mid \theta \,;\, B(\mu B)\vec{t} \mid \Xi', B_2 \mid \Xi_2, \ldots, B_m \mid \Xi_m \,;\, nat\ s \rangle.$$

– If $B_1$ is $\exists y.\ B\ y$ then the last inference rule of $\Xi_1$ is $\exists$. Let $r$ be the substitution term used to introduce this $\exists$ quantifier and let $\Xi'$ be the proof of the premise of that inference rule. Then we make a transition to

$$\langle \Sigma, w \mid \theta \circ \varphi \,;\, B\ w \mid \Xi', B_2 \mid \Xi_2, \ldots, B_m \mid \Xi_m \,;\, nat\ s \rangle,$$

where $w$ is a variable not in $\Sigma$ and $\varphi$ is the substitution $[w \mapsto r]$. Here, we assume that the composition of substitutions satisfies the equation $(\theta \circ \varphi)(x) = \varphi(\theta(x))$.

– If $B_1$ is $u = v$ and the terms $u$ and $v$ are unifiable with most general unifier $\varphi$, then we make a transition to $\langle \Sigma\varphi \mid \rho \,;\, \varphi(B_2) \mid \Xi_2, \ldots, \varphi(B_m) \mid \Xi_m \,;\, nat\ (\varphi t) \rangle$ where $\rho$ is the substitution such that $\theta = \varphi \circ \rho$.

In each of these cases, we must show that the transition is made to an augmented state. In all but the last two rules above, this is easy to show. In the case of the transition due to $\exists$, we know that $\Xi'$ is a proof of $\theta(B\ r)$ but that formula is simply $\varphi(\theta(B\ w))$ since $W$ is new and $r$ contains no variables free in $\Sigma$. In the case of the transition due to equality, we know that $\Xi_1$ is a proof of the formula $\theta(u = v)$ which means that $\theta u$ and $\theta v$ are the same term and, hence, that $u$ and $v$ are unifiable and that $\theta$ is a unifier. Let $\varphi$ be the most general unifier of $u$ and $v$. Thus, there is a substitution $\rho$ such that $\theta = \varphi \circ \rho$ and, for $i \in \{2, \ldots, m\}$, $\Xi_i$ is a proof of $(\varphi \circ \rho)(B_i)$.

Finally, termination of this algorithm is ensured since the number of occurrences of inference rules in the included proofs decreases at every step of the transition. Since we have shown that there is an augmented path that terminates, we have that there exists a path of states to a success state with value $t$.

Thus, a (naive) proof-search algorithm involving both unification and non-deterministic search is sufficient for computing the functions encoded in relations. We will see in Section 8 that we can capture at least those functions that are provably total in a system analogous to $I\Sigma_1$. This is the class of primitive recursive functions, and [3] gives a coding for them in $\mu$MALL.

While it is easy to encode the proof of totality for the Ackermann function in $\mu$LK, it seems unlikely that a totality proof for that function can be done within $\mu$MALL. This separation between $\mu$LK and $\mu$MALL was conjectured by Baelde [3, Section 3.5]. There are also a number of other linear logic style systems for which the totality of Ackermann's function is known to be not provable. In particular, if we developed a Curry-Howard interpretation of $\mu$MALL, it would yield a system close to the linear $\lambda$-terms $H(\emptyset)$ of [26], which is known to capture exactly primitive recursive functions (see also similar results in [25]).

## 7 $\mu$LK and Peano Arithmetic

In this section, we show that provable statements in Peano arithmetic are also provable in $\mu$LK. Since induction in our proof systems is performed as introductions on

fixed point formulas and not on (inductive) types, we introduce the following translation function that inserts the inductive predicate *nat* into all quantified formulas. First, recall that in the Church's STT representation of quantified formulas, the universally quantified formula $\forall x.B$ is an abbreviation for $\forall(\lambda x.B)$: here, $\forall$ is a constant of type $(\iota \to o) \to o$. (The existential quantifier is similarly coded by the constant $\exists$ of the same type.) Second, we denote by $Q^\circ$ the result of replacing every occurrence of $\forall$ with $\lambda B.\forall x\,(\overline{nat\ x} \parr (Bx))$ and every occurrence of $\exists$ with $\lambda B.\exists x\,(nat\ x \otimes (Bx))$. Also, Peano Arithmetic is usually presented as the system consisting of classical logic with equality, and the additional constants $s, z, +, \cdot$ and the following axioms.

$$\begin{array}{ll}
\forall x.\ (s\,x) \neq \mathbf{0} & \forall x.\ \forall y(x + s\,x) = s(x + y) \\
\forall x \forall y.\ (s\,x = s\,y) \supset (x = y) & \forall x.\ (x \cdot \mathbf{0} = \mathbf{0}) \\
\forall x.\ (x + \mathbf{0} = x) & \forall x.\ (x \cdot s\,y = (x \cdot y + x)) \\
(A\mathbf{0} \wedge \forall x.\ (Ax \supset A(s\,x))) \supset \forall x.\, Ax &
\end{array}$$

Since we wish to avoid introducing the extra constructors + and $\cdot$, we encode addition and multiplications as relations. We can then extend the translation $(\cdot)^\circ$ to include

$$(x + y = w)^\circ := plus\,x\,y\,w \qquad \text{and} \qquad (x \cdot y = w)^\circ := mult\,x\,y\,w,$$

where *mult* is the expression

$$\mu\lambda M\lambda n\lambda m\lambda p((n = \mathbf{0} \otimes p = \mathbf{0}) \oplus \exists n'\exists p'(n = (s\,n') \otimes plus\ m\ p'\ p \otimes M\ n'\ m\ p')).$$

**Theorem 4 ($\mu$LK contains Peano arithmetic).** *Let $Q$ be any unpolarized formula and let $\hat{Q}$ be a polarized version of $Q$. If $Q$ is provable in Peano arithmetic then $(\hat{Q})^\circ$ is provable in $\mu$LK.*

*Proof.* It is easy to prove that *mult* and *plus* describe precisely the multiplication and addition operations on natural numbers. Furthermore, the translations of the Peano Axioms can all be proved in $\mu$LK. We illustrate just one of these axioms here. In particular, a polarization of the translation of the induction scheme is

$$\overline{(A\mathbf{0} \otimes \forall x.\ (\overline{nat\ x} \parr \overline{Ax} \parr A(s\,x)))} \parr \forall x.\ (\overline{nat\ x} \parr Ax)$$

An application of the $\nu$ rule to the second occurrence of $\overline{nat\ x}$ can provide an immediate proof of this axiom. Finally, the cut rule in $\mu$LK allows us to encode the inference rule of modus ponens.

## 8 Linearized arithmetic

We now consider $\mu$MALL and $\mu$LK proofs at level 3, i.e., with no restrictions on the occurrences the $\mu\nu$ rule and the fixed point rules. We will, however, consider restrictions on the nesting of polarities within theorems and invariants.

When trying to compare $\mu$MALL with $\mu$LK, we find that there are a number of statements that are provable in $\mu$LK but not in $\mu$MALL since their proofs require contraction. Take for example the formula $\bot \multimap \bot \otimes \bot$ or the formula $\forall x \forall y(x = y \oplus x \neq y)$

(mentioned in Section 3), which are provable in $\mu$LK by using contraction, but not provable in $\mu$MALL. We now wish to identify classes of formulas for which provability in $\mu$LK is conservative over $\mu$MALL: we do this by making restrictions on the polarities of the connectives that appear in formulas. In Section 2, we defined the notions of *purely positive* and *purely negative* formulas. Both of the small counterexamples we mentioned above involved an alternation of positive and negative connectives. Our first conservativity results involves examining purely positive formulas.

**Theorem 5.** *Let $\Gamma$ be a multiset of purely positive formulas. If $\vdash \Gamma$ has a $\mu$LK proof, then there exists a $P \in \Gamma$ such that $\vdash P$ has a $\mu$MALL proof.*

*Proof.* This proof proceeds by induction on the structure of $\mu$LK proofs. Since the $\mu\nu$ rule is not applicable, the only possible base cases are the introduction rules for = and 1, and, in both cases, the theorem holds immediately.

In the inductive step, consider the case of an application of the $\otimes$ rule to derive the sequent $\vdash \Gamma, \Delta, P \otimes Q$ from the premises $\vdash \Gamma, P$ and $\vdash \Delta, Q$. By the inductive hypothesis, we can select from each of these premises a formula that is provable in $\mu$MALL. We distinguish three different subcases depending on which of the formulas in the premises are selected.

- $P$ and $Q$ are selected. Then by an application of $\otimes$ we can prove in $\mu$MALL $P \otimes Q$, which appears in the endsequent.
- $R \in \Gamma$ is selected. Then $R$ is provable in $\mu$MALL by inductive hypothesis and appears in the endsequent.
- $R \in \Delta$ is selected. As in the previous point.

The cases for introducing $\oplus, \exists, \mu$ are analogous and simpler. We are left with the cases of weakening and contraction and, in these cases, the conclusion is immediate.

A consequence of this theorem and Theorem 4 is the following.

**Corollary 1.** *Let $Q$ be an unpolarized formula that can be polarized as a purely positive formula $\hat{Q}$. If $\vdash Q$ has a proof in Peano Arithmetic then $\vdash \hat{Q}^\circ$ has a proof in $\mu$MALL.*

It is important to note that induction (the $\nu$ inference rule) does not play a role in this proof, since it would require the presence of the negative connective $\nu$.

The use of induction in $\mu$MALL allows for some formulas to be weakened and contracted. The following proposition is well known and can be proved by induction of the structure of purely negative formulas [4].

**Proposition 1.** *The weakening and contraction rules are admissible in $\mu$MALL for purely negative formulas.*

Another way to state this proposition is that the linear logic equivalence $\underline{N \multimap ?N}$ holds for purely negative formulas $N$. Thus, expressions such as $\underline{\mathit{nat}\ \mathbf{5}}$ and $\underline{\mathit{plus}\ n\ m\ p}$ can be used any number of times within a $\mu$MALL proof. (If we presented a two-sided sequent system for $\mu$MALL then assumptions such as $\mathit{nat}\ \mathbf{5}$ and $\mathit{plus}\ n\ m\ p$ can be used any number of times.) As a result, $\mu$MALL proofs can, occasionally, resemble proofs in a classical logic setting. A similar result is known for linear $\lambda$-calculi, where, in the presence of a recursor, one can obtain duplication and erasure for the type of natural numbers (but not for functions) [1].

**Definition 1.** *A* n/p formula *is a negative formula in which no negative connective occurrence appears in the scope of a positive connective. Thus, a* n/p *formula consists of some negative top-level connectives with purely positive subformulas underneath: there is at most one alternation of polarity from negative to positive.*

By employing the translation from Peano Arithmetic introduced in Section 7, we can view these classes as containing some fragments of the arithmetical hierarchy.

**Proposition 2.** *Let P be a formula of Peano Arithmetic. Then*

- *If P is* $\Sigma_1^0$*, there is a polarization* $\hat{P}^\circ$ *that is purely positive.*
- *If P is* $\Pi_2^0$*, there is a polarization* $\hat{P}^\circ$ *that is a* n/p *formula.*

One should not be led into confusion by thinking that, thanks to Theorem 5, we could prove open, purely positive formulas and then strengthen them by universal quantification to get stronger theorems expressing, for example, the totality of a function. For example, a formula such as $\exists x\,.\mathit{plus}\,a\,b\,x$ is not provable since it requires more information on the $a$ and $b$ variables (and the proof needs to proceed by induction on them). The provable formula that expresses totality of the plus relation is then $\forall x.\,\mathit{nat}\,x \supset \forall y.\,\mathit{nat}\,y \supset \exists u\,.\mathit{plus}\,x\,y\,u$, which is a n/p formula. Note that the two examples of $\mu$LK provable formulas without $\mu$MALL proofs at the beginning of this section are not n/p formulas.

When the induction rule $\nu$ is available, we will restrict occurrences of inductive invariants to be purely positive in order to prevent complex formulas from appearing in proofs. We call $\mu\mathrm{LK}_1$ the system consisting of the same rules as $\mu$LK but where the inductive invariants are restricted to be purely positive. The notation comes from the fact that this fragment is similar to the fragment $I\Sigma_1$ of Peano Arithmetic.

We now finish this section with proving that any n/p formula provable in $\mu\mathrm{LK}_1$ is provable in $\mu$MALL. This conservativity result can be applied to the formulas stating the *totality* and *determinancy* properties (see Section 6) of relations defined by purely positive fixed points, since they are all n/p formulas. The proof of this result would be aided greatly if we had a focusing theorem for $\mu$LK. If we take the focused proof system for $\mu$MALL given in [4,6] and add contraction and weakening in the usual fashion, we have a natural candidate for a focused proof system for $\mu$LK. However, the completeness of that proof system is currently open. As Girard points out in [17], the completeness of such a focused (cut-free) proof system would allow the extraction of the constructive content of classical $\Pi_2^0$ theorems, and we should not expect such a result to follow from the usual ways that we prove cut-elimination and the completeness of focusing. As a result of not possessing such a focused proof system for $\mu$LK, we must reproduce aspects of focusing in order to prove our conservation result.

**Definition 2.** *A* reduced sequent *is a sequent that contains only purely negative, purely positive, and* n/p *formulas. If* $\Gamma_1$ *and* $\Gamma_2$ *are reduced sequents, we say that* $\Gamma_1$ contains $\Gamma_2$ *if* $\Gamma_2$ *is a sub-multiset of* $\Gamma_1$*. Finally, we say that a reduced sequent is a* pointed *sequent if it contains exactly one formula that is either purely positive or* n/p*.*

**Definition 3.** *A* positive region *is a cut-free* $\mu LK_1$ *proof that contains only the inference rules* $\mu\nu$*, contractions, weakening, and introduction rules for the positive connectives.*

**Definition 4.** *The Cνν rule is the following derived rule of inference.*

$$\frac{\vdash \Gamma, S\vec{t}, U\vec{t} \quad \vdash BU\vec{x}, \overline{U\vec{x}} \quad \vdash BS\vec{x}, \overline{S\vec{x}}}{\vdash \Gamma, \nu B\vec{t}}\ C\nu\nu$$

The $C\nu\nu$ rule is justified as the following combination of $\nu$ and contraction rules.

$$\dfrac{\dfrac{\dfrac{\vdash \Gamma, S\vec{t}, U\vec{t} \quad \vdash BU\vec{x}, \overline{U\vec{x}}}{\vdash \Gamma, \nu B\vec{t}, S\vec{t}}\ \nu \qquad \vdash BS\vec{x}, \overline{S\vec{x}}}{\vdash \Gamma, \nu B\vec{t}, \nu B\vec{t}}\ \nu}{\vdash \Gamma, \nu B\vec{t}}\ C$$

Since we are working within $\mu\mathrm{LK}_1$, the invariants $S$ and $U$ are purely positive.

**Definition 5.** *A* negative region *is a cut-free $\mu LK_1$ partial proof in which the open premises are all reduced sequent and where the only inference rules are introductions for negative connectives plus the Cνν rule.*

**Lemma 1.** *If a reduced sequent $\Gamma$ has a positive region proof then $\Gamma$ contains a pointed sequent that has a $\mu$MALL proof.*

*Proof.* This proof is a simple generalization of the proof of Theorem 5.

**Lemma 2.** *If every premise of a negative region contains a pointed sequent with a $\mu$MALL proof, then the conclusion of the negative region contains a pointed sequent with a $\mu$MALL proof.*

*Proof.* This proof is by induction on the height of the negative region. The most interesting case to examine is the one where the last inference rule of the negative region is the $C\nu\nu$ rule. Referring to the inference rule displayed above, the inductive hypothesis ensures that the reduced sequent $\vdash \Gamma, S\vec{t}, U\vec{t}$ contains a pointed sequent $\Delta, C$ where $\Delta$ is a multiset of purely negative formula in $\Gamma$ and where the formula $C$ (that is either purely positive or is $\mathtt{n/p}$) is either a member of $\Gamma$ or is equal to either $S\vec{t}$ or $U\vec{t}$. In the first case, $\Delta, C$ is also contained in the endsequent $\Gamma, \nu B\vec{t}$. In the second case, we have one of the following proofs:

$$\frac{\vdash \Delta, S\vec{t} \quad \vdash BS\vec{x}, \overline{S\vec{x}}}{\vdash \Gamma, \nu B\vec{t}}\ \nu \qquad \frac{\vdash \Delta, U\vec{t} \quad \vdash BU\vec{x}, \overline{U\vec{x}}}{\vdash \Gamma, \nu B\vec{t}}\ \nu$$

depending on whether or not $C$ is $S\vec{t}$ or $U\vec{t}$.

**Lemma 3.** *If the reduced sequent $\Gamma$ has a cut-free $\mu LK_1$ proof then $\Gamma$ has a proof that can be divided into a negative region that proves $\Gamma$ in which all its premises have positive region proofs.*

*Proof.* This lemma is proved by appealing to the permutation of inference rules. As shown in [4], the introduction rules for negative connectives permute down over all inference rules in $\mu$MALL. Not considered in that paper is how such negative introduction

rules permute down over contractions. It is easy to check that such permutations do, in fact, happen except in the case of the $\nu$ rule. In general, contractions below a $\nu$ rule will not permute upwards, and, as a result, the negative region is designed to include the $C\nu\nu$ rule (where contraction is stuck with the $\nu$ rule). As a result, negative rules (including $C\nu\nu$) permute down while contraction and introductions of positive connectives permute upward. This gives rise to the two-region proof structure.

By combining the results of this section we get the following comparison between classical and linear arithmetic:

**Theorem 6.** *Any* $\mathtt{n/p}$ *formula provable in* $\mu LK_1$ *is provable in* $\mu MALL$.

Totality statements for primitive recursive functions are included among the formulas to which Theorem 6 can be applied. Such totality statements provide a lower bound for what can be accomplished by computations in the style of Theorem 3. Recent work, such as [25], shows how allowing a more general induction rule in a linear setting, where the context is duplicated and provided to the right branch, makes it possible to capture computations that go beyond primitive recursion. Such an extension to $\mu$MALL would increase the extent of Theorem 3, although its relation to fragments of $\mu$LK remains to be assessed.

The *focused inductive theorem proving* strategy reported in [7] is justified (in large part) by Theorem 6. Being able to rule out, for example, the need to use the contraction rule within proof search procedures can greatly restrict the search space for proofs.

In this paper, we have stayed rather close to a traditional presentation of sequent calculus proofs as finite tree structures. However, a good deal of recent research into the proof theory of arithmetic has gone into developing approaches to cyclic and infinite proof structures. A natural next step for the work described here is to consider possible connections between infinitary and cyclic proof systems in the style of CLKID$^\omega$ of [8] or $\mu MALL^\infty$ of [5].

## 9 Conclusion

We have used $\mu$MALL as a linearized approach to arithmetic. We have shown that various computational logic topics—generalized unification problems, Horn clause provability, model checking queries, computation of general recursive functions—can be captured correctly within this linearized arithmetic. Thus, these activities can be considered as relying on linear and not classical proof theory principles. We have shown also that the notion of polarity, that first arose in linear logic, has an interesting connection with the familiar notion of arithmetical hierarchy. Such a refinement to the proof theory of Peano arithmetic should be a significant aid when one turns to implementing proof search in arithmetic since ruling out the contraction rule often greatly constrains the search space. We plan to further develop the proof theories of $\mu$MALL and $\mu$LK particularly as this might provide a linearized version of the *reverse mathematics* project of Friedman [14].

**Acknowledgments:** We thank the anonymous reviewers of earlier versions of this paper for their comments.

## References


1. Sandra Alves, Maribel Fernández, Mário Florido, and Ian Mackie. The power of linear functions. In *CSL 2006: Computer Science Logic*, pages 119–134, 2006. doi:10.1007/11874683_8.
2. Peter B. Andrews. Provability in elementary type theory. *Zeitschrift fur Mathematische Logic und Grundlagen der Mathematik*, 20:411–418, 1974.
3. David Baelde. *A linear approach to the proof-theory of least and greatest fixed points*. PhD thesis, Ecole Polytechnique, December 2008. URL: http://arxiv.org/abs/0910.3383.
4. David Baelde. Least and greatest fixed points in linear logic. *ACM Trans. on Computational Logic*, 13(1):2:1–2:44, April 2012. doi:10.1145/2071368.2071370.
5. David Baelde, Amina Doumane, and Alexis Saurin. Infinitary proof theory: the multiplicative additive case. In Jean-Marc Talbot and Laurent Regnier, editors, *25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France*, volume 62 of *LIPIcs*, pages 42:1–42:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.CSL.2016.42.
6. David Baelde and Dale Miller. Least and greatest fixed points in linear logic. In N. Dershowitz and A. Voronkov, editors, *International Conference on Logic for Programming and Automated Reasoning (LPAR)*, volume 4790 of *LNCS*, pages 92–106, 2007. doi:10.1007/978-3-540-75560-9_9.
7. David Baelde, Dale Miller, and Zachary Snow. Focused inductive theorem proving. In J. Giesl and R. Hähnle, editors, *Fifth International Joint Conference on Automated Reasoning*, number 6173 in LNCS, pages 278–292, 2010. doi:10.1007/978-3-642-14203-1_24.
8. James Brotherston and Alex Simpson. Sequent calculi for induction and infinite descent. *J. of Logic and Computation*, 21(6):1177–1216, December 2011.
9. Alonzo Church. A formulation of the Simple Theory of Types. *J. of Symbolic Logic*, 5:56–68, 1940. doi:10.2307/2266170.
10. K. L. Clark. Negation as failure. In J. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, New York, 1978. doi:10.1007/978-1-4684-3384-5_11.
11. Hubert Comon and Pierre Lescanne. Equational problems and disunification. *Journal of Symbolic Computation*, 7:371–425, 1989.
12. Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. The structure of exponentials: Uncovering the dynamics of linear logic proofs. In Georg Gottlob, Alexander Leitsch, and Daniele Mundici, editors, *Kurt Gödel Colloquium*, volume 713 of *LNCS*, pages 159–171. Springer, 1993.
13. Roy Dyckhoff. Contraction-free sequent calculi for intuitionistic logic. *J. of Symbolic Logic*, 57(3):795–807, September 1992. doi:10.2307/2275431.
14. Harvey Friedman. Some systems of second order arithmetic and their use. In *Proceedings of the International Congress of Mathematicians (Vancouver, B. C., 1974), Vol. 1*, pages 235–242. Canad. Math. Congress, Montreal, Que., 1975.
15. Vijay Gehlot and Carl Gunter. Normal process representatives. In *5th Symp. on Logic in Computer Science*, pages 200–207, Philadelphia, Pennsylvania, June 1990. IEEE Computer Society Press. doi:10.1109/LICS.1990.113746.
16. Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–102, 1987. doi:10.1016/0304-3975(87)90045-4.
17. Jean-Yves Girard. A new constructive logic: classical logic. *Math. Structures in Comp. Science*, 1:255–296, 1991. doi:10.1017/S0960129500001328.
18. Jean-Yves Girard. A fixpoint theorem in linear logic. An email posting archived at https://www.seas.upenn.edu/~sweirich/types/archive/1992/msg00030.html to the linear@cs.stanford.edu mailing list, February 1992.

19. Jean-Yves Girard. Schrödinger's cut: La logique à la lumière du quantique. Unpublished., February 2021. URL: http://girard.perso.math.cnrs.fr/chat.pdf.
20. Lars Hallnäs and Peter Schroeder-Heister. A proof-theoretic approach to logic programming. II. Programs as definitions. *J. of Logic and Computation*, 1(5):635–660, October 1991. doi:10.1093/logcom/1.5.635.
21. Quentin Heath and Dale Miller. A proof theory for model checking. *J. of Automated Reasoning*, 63(4):857–885, 2019. doi:10.1007/s10817-018-9475-3.
22. Joshua Hodas and Dale Miller. Logic programming in a fragment of intuitionistic linear logic. *Information and Computation*, 110(2):327–365, 1994. doi:10.1006/inco.1994.1036.
23. William A. Howard. The formulae-as-type notion of construction, 1969. In J. P. Seldin and R. Hindley, editors, *To H. B. Curry: Essays in Combinatory Logic, Lambda Calculus, and Formalism*, pages 479–490. Academic Press, New York, 1980.
24. Naoki Kobayashi and Akinori Yonezawa. Asynchronous communication model based on linear logic. *Formal Aspects of Computing*, 7(2):113–149, 1995. doi:10.1007/BF01211602.
25. Denis Kuperberg, Laureline Pinault, and Damien Pous. Cyclic proofs, System T, and the power of contraction. In *Proceedings of the ACM on Programming Languages - POPL*, volume 5, pages 1–28, January 2021. doi:10.1145/3434282.
26. Ugo Dal Lago. The geometry of linear higher-order recursion. *ACM Trans. Comput. Log.*, 10(2):8:1–8:38, 2009. doi:10.1145/1462179.1462180.
27. Raymond McDowell, Dale Miller, and Catuscia Palamidessi. Encoding transition systems in sequent calculus. *Theoretical Computer Science*, 294(3):411–437, 2003. doi:10.1016/S0304-3975(01)00168-2.
28. Dale Miller. Unification under a mixed prefix. *Journal of Symbolic Computation*, 14(4):321–358, 1992. doi:10.1016/0747-7171(92)90011-R.
29. Dale Miller. The $\pi$-calculus as a theory in linear logic: Preliminary results. In E. Lamma and P. Mello, editors, *3rd Workshop on Extensions to Logic Programming*, number 660 in LNCS, pages 242–265, Bologna, Italy, 1993. Springer. URL: http://www.lix.polytechnique.fr/Labo/Dale.Miller/papers/pic.pdf.
30. Dale Miller. Forum: A multiple-conclusion specification logic. *Theoretical Computer Science*, 165(1):201–232, September 1996. doi:10.1016/0304-3975(96)00045-X.
31. Dale Miller and Alexandre Viel. Proof search when equality is a logical connective. *Annals of Mathematics and Artificial Intelligence*, July 2021. To appear in the *Special Issue on Theoretical and Practical Aspects of Unification*. doi:10.1007/s10472-021-09764-0.
32. Gopalan Nadathur. A proof procedure for the logic of hereditary Harrop formulas. *Journal of Automated Reasoning*, 11(1):115–145, August 1993. doi:10.1007/BF00881902.
33. Vivek Nigam and Dale Miller. Algorithmic specifications in linear logic with subexponentials. In António Porto and Francisco Javier López-Fraguas, editors, *ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP)*, pages 129–140. ACM, 2009. doi:10.1145/1599410.1599427.
34. Lawrence C. Paulson. The foundation of a generic theorem prover. *Journal of Automated Reasoning*, 5:363–397, September 1989. doi:10.1007/BF00248324.
35. Peter Schroeder-Heister. Rules of definitional reflection. In M. Vardi, editor, *8th Symp. on Logic in Computer Science*, pages 222–232. IEEE Computer Society Press, IEEE, June 1993. doi:10.1109/LICS.1993.287585.
36. Sten-Ake Tärnlund. Horn Clause Computability. *BIT*, 17:215–226, 1977.

## A Consistency of $\mu$LK

By second-order linear logic, *LL2*, we mean the logic of MALL with the addition of the following logical connectives: two exponentials ! and ?, negation $(\cdot)^\perp$, equality and non-equality, and first-order and second-order quantification (no occurrences of fixed points are permitted). Cut-elimination of this version of *LL2* follows from Girard's original cut-elimination proof [16] and the cut-elimination proofs known for equality and non-equality [18,35].

We translate $\mu$LK formulas into *LL2* formulas by translating fixed point expressions into second-order quantified formulas. The least fixed point expression $\mu B\vec{x}$ should be translated to a formula roughly of the form $\forall S\Big(\,!(\forall\vec{y}\ .\ BS\vec{y} \multimap S\vec{y}) \multimap S\vec{x}\,\Big)$. This translation must also insert ? into formulas in order to account for the fact that in $\mu$LK, any formula can be contracted and weakened at any point in a proof. The translation is given as follows.

- $\lceil t = s\rceil = ?(t = s)$ and $\lceil t \neq s\rceil = ?(t \neq s)$
- $\lceil\forall x.Px\rceil = ?\,\forall x.\lceil Px\rceil$ and $\lceil\exists x.Px\rceil = ?\,\exists x.\lceil Px\rceil$.
- $\lceil B \otimes C\rceil = ?(\lceil B\rceil \otimes \lceil C\rceil)$, $\lceil B \parr C\rceil = ?(\lceil B\rceil \parr \lceil C\rceil)$, $\lceil B \,\&\, C\rceil = ?(\lceil B\rceil \,\&\, \lceil C\rceil)$, $\lceil B \oplus C\rceil = ?(\lceil B\rceil \oplus \lceil C\rceil)$
- $\lceil 1\rceil = ?\,1$, $\lceil\perp\rceil = ?\,\perp$, $\lceil 0\rceil = ?\,0$, $\lceil\top\rceil = ?\,\top$
- $\lceil\mu B\vec{x}\rceil = ?\,\forall S\,[(?\,\exists\vec{y}\,.\,\lceil B\rceil S\vec{y} \otimes (S\vec{y})^\perp) \parr S\vec{x}\,]$
- $\lceil\nu B\vec{x}\rceil = ?\,\exists S\,[(!\,\forall\vec{y}\,.\,\lceil B\rceil S\vec{y} \parr (S\vec{y})^\perp) \otimes S\vec{x}\,]$
- $\lceil A\rceil = A$ where $A$ is an atomic formula.
- The $\lceil\cdot\rceil$ operator commutes with $\lambda$-abstraction: $\lceil\lambda x.B\rceil = \lambda x.\lceil B\rceil$. This feature of $\lceil\cdot\rceil$ permit translating invariants and the body of fixed point expressions.
- The $\lceil\cdot\rceil$ operator can be applied to a multiset of formulas: $\lceil\Gamma\rceil = \{\lceil P\rceil \mid P \in \Gamma\}$.

Note that when $B$ is the $\lambda$-abstraction $\lambda p\lambda\vec{x}.C$, where $C$ is a $\mu$MALL formula, $p$ is a first-order predicate variable, and $\vec{x}$ is a list of first-order terms, then $\lceil B\rceil\lceil S\,\rceil\vec{t}$ is equal to $\lceil BS\,\vec{t}\,\rceil$ up to $\lambda$-conversion.

We shall also need the following inference rule in *LL2*, which is a kind of generalization of the cut rule.

$$\frac{\vdash \Gamma, BQ\vec{t} \quad \vdash \neg(Q\vec{x}), P\vec{x}}{\vdash \Gamma, BP\vec{t}}\ deep.$$

Here, of course, the first-order variables $\vec{x}$ are new. Also, the expression $B$ has the type that takes a first-order predicate to a first-order predicate and also *monotonic*, meaning that there are no occurrences of negated predicate variables in $B$. It is proved in [6, Proposition 2] that this rule is admissible in *LL2*. This rule essentially allows us to move from the fact that $Q \subseteq P$ and to the fact that $BQ \subseteq BP$.

**Lemma 4.** *If $\vdash \Gamma$ is derivable in $\mu$LK then $\vdash \lceil\Gamma\rceil$ is derivable in LL2.*

*Proof.* We proceed by induction on the structure of cut-free $\mu$LK proofs. In particular, assume that $\vdash \Gamma$ has a cut-free $\mu$LK proof $\Xi$.

**Case:** The last inference rule of $\Xi$ comes from Figure 1, *i.e.*, it is an introduction rules for a propositional connective, a unit, or a quantifier. For example, assume that this last inference rule is the following $\otimes$ introduction rule.

$$\frac{\vdash \Gamma, P \quad \vdash \Delta, Q}{\vdash \Gamma, \Delta, P \otimes Q}\ \otimes$$

By the inductive assumption, $\vdash \lceil\Gamma\rceil, \lceil P\rceil$ and $\vdash \lceil\Delta\rceil, \lceil Q\rceil$ have *LL*2 proofs. Hence, $\vdash \lceil\Gamma\rceil, \lceil\Delta\rceil, \lceil P\rceil \otimes \lceil Q\rceil$ has an *LL*2 proof. By using the dereliction rule for ? and the definition of $\lceil\cdot\rceil$, we know that $\vdash \lceil\Gamma, \Delta\rceil, \lceil P \otimes Q\rceil$ has an *LL*2 proof.

**Case:** The last inference rule is either weakening $W$ or contraction $C$. Since the image of $\lceil\cdot\rceil$ always has a ? exponential as its top-level connective, the corresponding *LL*2 inference rule is built with the same structural rule.

**Case:** The last inference rule of $\Xi$ is one of the fixed point rules from Figure 2. Assume, for example, that the last rule is

$$\frac{}{\vdash \mu B\vec{t}, \nu\overline{B}\vec{t}}\ \mu\nu$$

The desired translation of this inference rule into *LL*2 is

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdash \lceil B\rceil S\vec{y}, \lceil\overline{B}\rceil(\lambda\vec{w}(S\vec{w})^{\perp})\vec{y} \quad \dfrac{}{\vdash (S\vec{y})^{\perp}, \neg((S\vec{x})^{\perp})}\ init}{\vdash \lceil B\rceil S\vec{y} \otimes (S\vec{y})^{\perp}, \lceil\overline{B}\rceil(\lambda\vec{w}(S\vec{w})^{\perp})\vec{y} \parr \neg((S\vec{y})^{\perp})}\ \parr, \otimes}{\vdash ?\exists\vec{y}.\lceil B\rceil S\vec{y} \otimes (S\vec{y})^{\perp}, !(\forall\vec{y}.\lceil\overline{B}\rceil(\lambda\vec{w}(S\vec{w})^{\perp})\vec{y} \parr \neg((S\vec{y})^{\perp}))}\ !R, ?D, \forall, \exists \quad \dfrac{}{\vdash S\vec{x}, (S\vec{x})^{\perp}}\ init}{\vdash ?(\exists\vec{y}.\lceil B\rceil S\vec{y} \otimes (S\vec{y})^{\perp}) \parr S\vec{x}, !(\forall\vec{y}.\lceil\overline{B}\rceil(\lambda\vec{w}(S\vec{w})^{\perp})\vec{y} \parr \neg((S\vec{y})^{\perp})) \otimes (S\vec{x})^{\perp}}\ \parr, \otimes}{\vdash ?(\exists\vec{y}\,.\,\lceil B\rceil S\vec{y} \otimes (S\vec{y})^{\perp}) \parr S\vec{x}, \exists S\,[!(\forall\vec{y}\,.\,\lceil\overline{B}\rceil S\vec{y} \parr (S\vec{y})^{\perp}) \otimes S\vec{x}\,]}\ \exists}{\vdash ?\forall S\,[?(\exists\vec{y}\,.\,\lceil B\rceil S\vec{y} \otimes (S\vec{y})^{\perp}) \parr S\vec{x}\,], ?\exists S\,[!(\forall\vec{y}\,.\,\lceil\overline{B}\rceil S\vec{y} \parr (S\vec{y})^{\perp}) \otimes S\vec{x}\,]}\ ?D, \forall$$

An induction on the structure of the formula $B$ provides a proof that there is an *LL*2 proof of remaining open premise.

Assume instead that the last rule of $\Xi$ is the introduction for $\nu$, namely,

$$\frac{\vdash \Gamma, S\vec{t} \quad \vdash BS\vec{x}, \overline{(S\vec{x})}}{\vdash \Gamma, \nu B\vec{t}}\ \nu.$$

The higher-order quantifier that appears in the *LL*2 encoding is instantiated with $\lceil S\rceil$. Thus, the desired *LL*2 proof is

$$\dfrac{\dfrac{\dfrac{\vdash \lceil BS\vec{x}\rceil, \lceil\overline{S\vec{x}}\rceil \quad \vdash \neg(\lceil\overline{S\vec{x}}\rceil), \neg(\lceil S\rceil\vec{x})}{\vdash \lceil B\rceil\lceil S\rceil\vec{x}, \neg(\lceil S\rceil\vec{x})}\ cut \qquad \vdash \lceil\Gamma\rceil, \lceil S\vec{t}\rceil}{\vdash \lceil\Gamma\rceil, !(\forall\vec{y}\,.\,\lceil B\rceil\lceil S\rceil\vec{y} \parr \neg(\lceil S\rceil\vec{y})) \otimes \lceil S\rceil\vec{t}}\ \otimes, \forall, \parr}{\vdash \lceil\Gamma\rceil, ?\exists S\,[!(\forall\vec{y}\,.\,\lceil B\rceil S\vec{y} \parr \neg(S\vec{y})) \otimes S\vec{t}\,]}\ ?D, \exists\, S \mapsto \lceil S\rceil$$

By the inductive hypothesis, the leftmost and rightmost premises have *LL*2 proof. Induction on first-order abstractions such as $S$ shows that the middle premise also has an *LL*2 proof.

Assume instead that the last rule of $\Xi$ is the introduction for $\mu$, namely,

$$\frac{\vdash \Gamma, B(\mu B)\vec{t}}{\vdash \Gamma, \mu B\vec{t}}\ \mu.$$

We first show that $\vdash \lceil B\rceil\lceil \mu B\rceil\vec{t} \multimap \lceil \mu B\vec{t}\rceil$ has an *LL*2 proof for all $B$ and $\vec{t}$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \overline{\lceil B\rceil\lceil \mu B\rceil\vec{t}}, \lceil B\rceil\lceil \mu B\rceil\vec{t}}\ init \qquad \Xi \vdash ?(\exists \vec{y}\ .\ \lceil B\rceil S\,\vec{y} \otimes (S\,\vec{y})^{\perp}), \overline{\lceil \mu B\rceil\vec{x}}, S\,\vec{x}}{\vdash \overline{\lceil B\rceil\lceil \mu B\rceil\vec{t}}, ?(\exists \vec{y}\ .\ \lceil B\rceil S\,\vec{y} \otimes (S\,\vec{y})^{\perp}), \lceil B\rceil S\,\vec{t}}\ deep \qquad \dfrac{}{\vdash (S\,\vec{t})^{\perp}, S\,\vec{t}}\ init}{\vdash \overline{\lceil B\rceil\lceil \mu B\rceil\vec{t}}, ?(\exists \vec{y}\ .\ \lceil B\rceil S\,\vec{y} \otimes (S\,\vec{y})^{\perp}), \lceil B\rceil S\,\vec{t} \otimes (S\,\vec{t})^{\perp}, S\,\vec{t}}\ \otimes}{\vdash \overline{\lceil B\rceil\lceil \mu B\rceil\vec{t}}, ?(\exists \vec{y}\ .\ \lceil B\rceil S\,\vec{y} \otimes (S\,\vec{y})^{\perp}), S\,\vec{t}}\ C, D, \exists}{\vdash \overline{\lceil B\rceil\lceil \mu B\rceil\vec{t}}, \lceil \mu B\vec{t}\rceil}\ ?, \forall, \parr$$

Here, $\Xi$ is a straightforward *LL*2 proof. Finally, using this proof of $\vdash \overline{\lceil B\rceil\lceil \mu B\rceil\vec{t}}, \lceil \mu B\vec{t}\rceil$ and the cut rule for *LL*2, we have shown the soundness of the $\mu$ rule in Figure 2.

*Proof (of Theorem 1).* Assume that $\vdash B$ and $\vdash \overline{B}$ have $\mu$LK proofs. By Lemma 4, we know that $\vdash \lceil B\rceil$ and $\vdash \lceil \overline{B}\rceil$ have *LL*2 proofs. While it is not the case that $\lceil \overline{B}\rceil = (\lceil B\rceil)^{\perp}$, a simple induction on the structure of $B$ shows that $\lceil \overline{B}\rceil \vdash (\lceil B\rceil)^{\perp}$ is provable in *LL*2. Since *LL*2 has a cut rule, we know that there is an *LL*2 proof of $\vdash \cdot$ (the empty sequent). By the cut-elimination theorem of *LL*2, this sequent also has a cut-free *LL*2 proof, which is impossible.