ISSN 1346-5597

National Institute of Informatics

NII Technical Report

Inhabitance of Existential Types is Decidable in Negation-Product Fragment

Makoto Tatsuta, Ken-etsu Fujita,
Ryu Hasegawa, and Hiroshi Nakano

NII-2008-005E
Apr. 2008

Inhabitance of Existential Types is Decidable in Negation-Product Fragment

Makoto Tatsuta[1], Ken-etsu Fujita[2], Ryu Hasegawa[3], and Hiroshi Nakano[4]

[1] National Institute of Informatics, Japan, `tatsuta@nii.ac.jp`
[2] Department of Computer Science, Gunma University, Japan
[3] Department of Mathematics, University of Tokyo, Japan
[4] Department of Applied Mathematics and Informatics, Ryukoku University, Japan

Abstract. This paper shows the inhabitance in the lambda calculus with negation, product, and existential types is decidable. This is proved by showing existential quantification can be eliminated and reducing the problem to provability in intuitionistic propositional logic. By the same technique, this paper also shows existential quantification followed by negation can be replaced by a specific witness in both that system and the system with implication and bottom.

1 Introduction

Existential types are important in computer science and have been studied actively for a long time. [6] showed types of abstract data types are existential types. [5] investigated a conversion translation by using existential types. Recently [2] showed there is a Galois correspondence between the polymorphic typed lambda calculus, that is the system F, and the type system with existential types. [4, 7, 8] tried to find a simple proof of strong normalization in type systems that contain existential types.

Inhabitance, type checking, and type inference of existential types are not known. For the system F, the inhabitance was proved to be undecidable [3, 11], and the type checking and the type inference were proved to be undecidable [12]. In addition, [1] proved the inhabitance is decidable in a type system with positive universal types. On the other hand, though existential types are dual to universal types, we do not know anything about inhabitance, type checking, nor type inference of existential types.

In this paper, we will investigate the type system with negation, product, and existential types proposed in [2], and show the inhabitance of that system is decidable. We will also show the inhabitance is also decidable when we add the bottom elimination rule to that system. These results are obtained by studying the corresponding logical systems and showing that their provability is decidable. In order to prove this, we will show the existential quantification elimination theorem, which says that any formula is equivalent to some formula without existential quantifiers. So the decidability of provability in these systems is reduced to that in intuitionistic propositional logic and minimal propositional logic. By using the same technique, we will also show that any formula with only existential quantification of negation is equivalent to some formula without existential quantification in the system with implication, existential quantification, and bottom. Moreover, in all these systems we will also show the second-order witness theorem, which states that we can replace existential quantification of negation by a specific witness effectively computed from the original formula.

The decidability of inhabitance in the negation-product-existential type system is interesting, because its dual system is the system F whose inhabitance is undecidable [3, 11]. This system can interpret the system F by CPS-translation, and this translation gives a Galois correspondence between the system F and this system [2]. So this system is closely related to the system F and has expressive power. However, at last this system will be shown to be decidable.

Our proof will use the following three key facts. The first one is that $A[X := \neg A[X := \mathrm{T}]]$ is equivalent to $\neg\neg A[X := \mathrm{T}]$ and $A[X := \mathrm{F}]$. This is proved by repeatedly using the property that $A[X := B]$ is equivalent to $A[X := \mathrm{T}]$ when B holds. The second one is that $\exists X \neg\neg A$ is equivalent to $\neg\neg A[X := \mathrm{T}] \vee \neg\neg A[X := \mathrm{F}]$. It is proved by combining the first one and $X \vee \neg X$. The third fact is that any formula is equivalent to conjunction of variables and a negation. It is proved by using the second fact. Existential quantification elimination follows from the second and the third facts.

As a byproduct of our proofs, in all these systems, we can also show the second-order witness theorem, which states $\exists X \neg A$ implies $\neg A[X := B]$ for some formula B effectively computed from the formula A itself. This is surprising because in first-order intuitionistic logic, the witness t such that $\neg A[x := t]$ holds can be obtained from the proof of $\exists x \neg A$ and it cannot be obtained from only the formula $\exists x \neg A$.

Section 2 defines the type system $\lambda\neg\wedge\exists$ with negation, product, and existential types, and the type system $\lambda\neg\wedge\exists\bot$ with negation, product, existential, and bottom types. The section also states decidability theorems of their inhabitance. Section 3 gives the corresponding logical systems $\mathrm{L}\neg\wedge\exists$ and $\mathrm{L}\neg\wedge\exists\bot$, and states decidability theorems of their provability as well as the existential quantification elimination theorem. The section also defines the logical system $\mathrm{L}\rightarrow\exists\bot$ with arrow, existential, and bottom types, and states its existential quantification elimination theorem. Section 4 discusses coding of logical connectives and double negation translation in minimal logic. Section 5 proves the existential quantification elimination theorems and the second-order witness theorem. In Section 6, we will prove the decidability theorems stated in Sections 2 and 3.

2 Existential Type Systems

We will define the type systems $\lambda\neg\wedge\exists$ and $\lambda\neg\wedge\exists\bot$, and state the decidability of inhabitance in these systems.

2.1 System $\boldsymbol{\lambda}\neg\wedge\exists$

We define the type system $\lambda\neg\wedge\exists$. We have variables $x, y, z, \ldots$ for λ-terms. λ-terms are defined by

$$M, N, \ldots ::= x|\lambda x.M|MM|\langle M, M\rangle|\pi_0 M|\pi_1 M|\langle\exists, M\rangle|M[x.M].$$

We have type variables $X, Y, Z, \ldots$. Types are defined by

$$A, B, \ldots ::= X|\bot|\neg A|A\wedge A|\exists XA.$$

Substitutions $M[x := N]$ and $A[X := B]$ are defined in a familiar way.

The system has the following inference rules:

$$\frac{\begin{array}{c}[x : A]\\ \vdots\\ M : \bot\end{array}}{\lambda x.M : \neg A}\,(\neg I) \qquad \frac{M : \neg A \quad N : A}{MN : \bot}\,(\neg E)$$

$$\frac{M : A \quad N : B}{\langle M, N\rangle : A\wedge B}\,(\wedge I) \qquad \frac{M : A\wedge B}{\pi_0 M : A}\,(\wedge E_1) \qquad \frac{M : A\wedge B}{\pi_1 M : B}\,(\wedge E_2)$$

$$\frac{M : A[X := B]}{\langle\exists, M\rangle : \exists XA}\,(\exists I) \qquad \frac{M : \exists XA \quad \begin{array}{c}[x : A]\\ \vdots\\ N : C\end{array}}{M[x.N] : C}\,(\exists E)$$

where the rule $(\exists E)$ has the standard variable condition.

Theorem 2.1 (Decidability of Inhabitance) *The inhabitance in the system $\lambda\neg\wedge\exists$ is decidable.*

Remark. This theorem says there is an algorithm such that for any types $A, B_1, \ldots, B_n$, this algorithm decides whether there is some term M such that $x_1 : B_1, \ldots, x_n : B_n \vdash M : A$ is provable.

2.2 System $\boldsymbol{\lambda}\neg\wedge\exists\bot$

We define the type system $\lambda\neg\wedge\exists\bot$. Its language is the same as that of $\lambda\neg\wedge\exists$. Its inference rules are those of $\lambda\neg\wedge\exists$ and

$$\frac{M : \bot}{M : A}\,(\bot E).$$

Theorem 2.2 (Decidability of Inhabitance) *The inhabitance in the system $\lambda\neg\wedge\exists\bot$ is decidable.*

We will prove these theorems in Section 6.

3 The Corresponding Logical Systems

We will investigate the corresponding logical systems L$\neg\wedge\exists$ and L$\neg\wedge\exists\perp$ to $\lambda\neg\wedge\exists$ and $\lambda\neg\wedge\exists\perp$ respectively. These systems are obtained from the corresponding type systems by dropping terms. We will state provability of those systems is decidable. We will also investigate a logical system L$\rightarrow\exists\perp$ and will state existential quantification followed by negation can be eliminated.

3.1 System L$\neg\wedge\exists$

We define the system L$\neg\wedge\exists$. We have second-order variables $X, Y, Z, \ldots$. We have the constant $\perp$. It does not have any meaning, since this system is based on minimal logic and does not have the bottom elimination rule. Formulas are the same as types of $\lambda\neg\wedge\exists$ and defined by $A, B, \ldots ::= X|\perp|\neg A|A \wedge A|\exists X A$.

Inference rules of the system are given by:

$$\dfrac{\begin{array}{c}[A]\\ \vdots \\ \perp\end{array}}{\neg A}\,(\neg I) \qquad \dfrac{\neg A \quad A}{\perp}\,(\neg E)$$

$$\dfrac{A \quad B}{A \wedge B}\,(\wedge I) \qquad \dfrac{A \wedge B}{A}\,(\wedge E_1) \qquad \dfrac{A \wedge B}{B}\,(\wedge E_2)$$

$$\dfrac{A[X := B]}{\exists X A}\,(\exists I) \qquad \dfrac{\exists X A \quad \begin{array}{c}[A]\\ \vdots \\ C\end{array}}{C}\,(\exists E)$$

where the rule $(\exists E)$ has the standard variable condition.

Two formulas A and B are defined to be equivalent if A is derivable from B and B is derivable from A.

Theorem 3.1 (Existential Quantification Elimination) *In* L$\neg\wedge\exists$, *any formula is equivalent to some formula without* $\exists$.

Theorem 3.2 (Decidability of Provability) *The provability in the system* L$\neg\wedge\exists$ *is decidable.*

3.2 System L$\neg\wedge\exists\perp$

We define the system L$\neg\wedge\exists\perp$. Its language is the same as that of L$\neg\wedge\exists$.

Its inference rules are those of L$\neg\wedge\exists$ and

$$\dfrac{\perp}{A}\,(\perp E).$$

Theorem 3.3 (Decidability of Provability) *The provability in the system* L$\neg\wedge\exists\perp$ *is decidable.*

Remark. The same statement as that of Theorem 3.1 holds in L$\neg\wedge\exists\perp$ since it is an extension of L$\neg\wedge\exists$.

3.3 System $\mathrm{L} \to \exists\bot$

We define the system $\mathrm{L} \to \exists\bot$. Formulas of this system is defined by

$$A, B, \ldots ::= X | \bot | A \to A | \exists X A.$$

The inference rules are $(\exists I)$, $(\exists E)$, $(\bot E)$, and

$$\frac{\begin{matrix}[A] \\ \vdots \\ B\end{matrix}}{A \to B}\ (\to I) \qquad \frac{A \to B \quad A}{B}\ (\to E)$$

Definition 3.4 A formula A is called an $\exists\neg$-formula if every occurrence of $\exists$ in A has the shape $\exists X \neg B$ for some X and B.

Theorem 3.5 (Existential Quantification Elimination) *In* $\mathrm{L} \to \exists\bot$, *any* $\exists\neg$*-formula is equivalent to some formula without* $\exists$.

We will prove these theorems in the rest of this paper.

4 Coding in Minimal Logic

We discuss some coding of logical connectives as well as a double negation translation in minimal logic.

In this section, we will discuss both $\mathrm{L}\neg \wedge \exists$ and $\mathrm{L} \to \exists\bot$ at the same time. All the statements in this section hold for both systems unless we explicitly specify a system.

First we define abbreviations for coding logical connectives. $\vee$, $\wedge$ and $\neg$ are coded in a standard way. We define F and T which mean the falsity and the truth.

Definition 4.1 In $\mathrm{L}\neg \wedge \exists$, we will use the following abbreviation.

$$A \vee B = \neg(\neg A \wedge \neg B).$$

In $\mathrm{L} \to \exists\bot$, we will use the following abbreviations.

$$\begin{aligned}&\neg A = A \to \bot, \\ &A \wedge B = (A \to B \to \bot) \to \bot, \\ &A \vee B = (A \to \bot) \to (B \to \bot) \to \bot.\end{aligned}$$

In both systems, we will use the following abbreviations.

$$\begin{aligned}&\mathrm{F} = \bot, \\ &\mathrm{T} = \neg\mathrm{F}.\end{aligned}$$

We will write J for one of $\mathrm{L}\neg \wedge \exists$ and $\mathrm{L} \to \exists\bot$. We will also write K for the classical logic obtained from J by adding the following rule $(\bot C)$.

$$\frac{\begin{matrix}[\neg A] \\ \vdots \\ \bot\end{matrix}}{A}\ (\bot C)$$

$A_1, \ldots, A_n \vdash_J B$ says that B is provable in the system J under the assumptions $A_1, \ldots, A_n$. $A_1, \ldots, A_n \dashv\vdash_J B_1, \ldots, B_m$ stands for $A_1, \ldots, A_n \vdash_J B_i$ for all i and $B_1, \ldots, B_m \vdash_J A_i$ for all i. We will sometimes write $\vdash$ and $\dashv\vdash$ for $\vdash_J$ and $\dashv\vdash_J$ respectively when they are not ambiguous.

We will write $A[B]$ for $A[X := B]$ when it is not ambiguous.

A context $\mathcal{C}$ is defined as a formula with the single hole $\cdot$ in a standard way. $\mathcal{C}[A]$ stands for variable-capturing substitution of a formula A for the hole in the context $\mathcal{C}$. We will use $\mathcal{C}$ to denote a context.

We prepare several basic lemmas. They are almost standard and given in [9, 10]. However, we will give some details since they are properties in second-order minimal logic without universal quantification when we discuss $\mathrm{L}\neg\wedge\exists$.

Lemma 4.2 *$A \dashv\vdash B$ implies $\mathcal{C}[A] \dashv\vdash \mathcal{C}[B]$.*

This is proved by induction on $\mathcal{C}$.

Lemma 4.3 *(1) $A \vdash \neg\neg A$.*
(2) $\neg A \dashv\vdash \neg\neg\neg A$.
(3) $\neg\neg(A \to B) \dashv\vdash \neg\neg A \to \neg\neg B$ in $\mathrm{L}\to\exists\bot$.
(4) $\neg\neg A \wedge \neg\neg B \dashv\vdash \neg\neg(A \wedge B)$.
(5) $\neg\neg\exists X A \dashv\vdash \neg\neg\exists X \neg\neg A$.
(6) $\vdash A \vee \neg A$.
(7) $\neg\neg A \vee \neg\neg B \dashv\vdash A \vee B$.
(8) $\neg\neg(A \vee B) \dashv\vdash A \vee B$.
(9) $A, B \vdash C$ implies $A \wedge B \vdash \neg\neg C$ in $\mathrm{L}\to\exists\bot$.

They are proved in a straightforward way.

The next lemma says that A and $A[\mathrm{T}]$ are equivalent when X holds, and $\neg A$ and $\neg A[\mathrm{F}]$ are equivalent when $\neg X$ holds.

Lemma 4.4 *(1) $X, A \dashv\vdash X, A[X := \mathrm{T}]$.*
(2) $\neg X, \neg\neg A \dashv\vdash \neg X, \neg\neg A[X := \mathrm{F}]$ in $\mathrm{L}\neg\wedge\exists$.
(3) $\neg X, A \dashv\vdash \neg X, A[X := \mathrm{F}]$ in $\mathrm{L}\to\exists\bot$.
(4) $A[X := \neg X], \neg\neg X \dashv\vdash A[X := \mathrm{F}], \neg\neg X$.

Proof. (1) By induction on A. We will discuss only an interesting case.

Case $\exists Y A_1$. By induction hypothesis, we have $X, A_1 \vdash A_1[\mathrm{T}]$. Hence we have $X, \exists Y A_1 \vdash \exists Y A_1[\mathrm{T}]$. Similarly we have $X, \exists Y A_1[\mathrm{T}] \vdash \exists Y A_1$.

(2) By induction on A. Cases are considered according to A. We will discuss only interesting cases.

Case X. The claim holds since $\mathrm{F} \dashv\vdash \neg\neg\mathrm{F}$ with $\neg X, \neg\neg X \vdash \mathrm{F}$ and $\mathrm{F} \vdash \neg\neg X$.

Case $A \wedge B$. By induction hypothesis and Lemma 4.3 (4).

Case $\exists Y A_1$. By induction hypothesis, we have $\neg X, \neg\neg A_1 \vdash \neg\neg A_1[\mathrm{F}]$. Hence we have $\neg X, \exists Y \neg\neg A_1 \vdash \exists Y \neg\neg A_1[\mathrm{F}]$. Therefore $\neg X, \neg\neg\exists Y \neg\neg A_1 \vdash \neg\neg\exists Y \neg\neg A_1[\mathrm{F}]$ holds. By Lemma 4.3 (5), we have $\neg X, \neg\neg\exists Y A_1 \vdash \neg\neg\exists Y A_1[\mathrm{F}]$. Similarly we have $\neg X, \neg\neg\exists Y A_1[\mathrm{F}] \vdash \neg\neg\exists Y A_1$.

(3) By induction on A. We use $\neg X, X \vdash \mathrm{F}$ and $\neg X, \mathrm{F} \vdash X$ for the case $A = X$. The other cases are proved in a similar way to (1).

(4) By induction on A. When A is X, the claim follows from $\mathrm{F} \vdash \neg X$. $\Box$

We give a definition of double negation translation.

Definition 4.5 (Double Negation Translation) For a formula A in J, we define the formula A^- by induction on A as follows:

$$\begin{aligned}
&X^- = \neg\neg X,\\
&\bot^- = \bot,\\
&(A \to B)^- = A^- \to B^-,\\
&(\neg A)^- = \neg A^-,\\
&(A \wedge B)^- = A^- \wedge B^-,\\
&(\exists X A)^- = \neg\neg\exists X A^-.
\end{aligned}$$

where the case $\to$ is for L$\to\exists\bot$ and the cases $\neg$ and $\wedge$ are for L$\neg\wedge\exists$.

Remark. (1) If A is a formula in J, then A^- is a formula in J.
(2) $(\neg A)^- = \neg A^-$ in L$\to\exists\bot$.

Lemma 4.6 *(1)* $\neg\neg A^- \dashv\vdash A^-$.
(2) $\mathcal{C}[A]^- \dashv\vdash \mathcal{C}^-[A^-]$ *where the hole* $\cdot$ *is interpreted by* $(\cdot)^- = \neg\neg(\cdot)$.
(3) $B_1, \ldots, B_n \vdash_K A$ *implies* $B_1^-, \ldots, B_n^- \vdash_J A^-$.
(4) $\neg\neg A \dashv\vdash A^-$.

This lemma is proved in the appendix.

The next lemma enables us to use classical reasoning in a negated context.

Lemma 4.7 $A \dashv\vdash B$ *in* K *implies* $\neg\mathcal{C}[A] \dashv\vdash \neg\mathcal{C}[B]$ *in* J.

This lemma is proved in the appendix.

We have Glivenko's theorem, which can be proved by double negation translation since we do not have universal quantification.

Proposition 4.8 $B_1, \ldots, B_n \vdash_K A$ *implies* $B_1, \ldots, B_n \vdash_J \neg\neg A$.

Proof. By Lemma 4.6 (3) and (4). □

5 Existential Quantification Elimination

We will prove the existential quantification theorems stated in Section 3. We will first show the existential quantification followed by negation can be eliminated in both L$\neg\wedge\exists$ and L$\to\exists\bot$. We will also show the second-order witness theorem in all these systems. In the first subsection, we will show the existential quantification can be eliminated and any formula with existential quantifiers is equivalent to some formula without existential quantifiers in L$\neg\wedge\exists$ and L$\neg\wedge\exists\bot$. In the second subsection, we will show existential quantification elimination for $\exists\neg$-formulas in L$\to\exists\bot$.

We will use a vector notation $\vec{e}$ to denote a sequence $e_1, \ldots, e_n$ $(n \geq 0)$. We will sometimes use $\vec{A}$ to denote a conjunction $A_1 \wedge \ldots \wedge A_n$ of formulas.

The next proposition is our key properties. The claim (1) is obtained by repeatedly replacing some formula by T when that formula holds. The claim (2) is a more readable form and immediately follows from (1). This claim is used for proving the claim (3) and the second-order witness theorem. The claim (3) says that an existential quantifier can be replaced by disjunction of the true case and the false case in the same way as in classical logic, when its body is negation.

Proposition 5.1 *In both* $\mathrm{L}\neg\wedge\exists$ *and* $\mathrm{L}\to\exists\bot$*, the following hold.*

(1) $A[X := \neg A[X := \mathrm{T}]] \dashv\vdash \neg\neg A[X := \mathrm{T}], A[X := \mathrm{F}]$.

(2) $\neg A[X := \neg A[X := \mathrm{T}]] \dashv\vdash \neg A[X := \mathrm{T}] \vee \neg A[X := \mathrm{F}]$.

(3) $\exists X\neg A \dashv\vdash \neg A[X := \mathrm{T}] \vee \neg A[X := \mathrm{F}]$.

Proof. (1) From the left-hand side to the right-hand side. By letting X be $\neg A[\mathrm{T}]$ in Lemma 4.4 (1), we have $A[\neg A[\mathrm{T}]], \neg A[\mathrm{T}] \vdash A[\mathrm{T}]$. Hence $A[\neg A[\mathrm{T}]] \vdash \neg\neg A[\mathrm{T}]$. By letting X be $A[\mathrm{T}]$ in Lemma 4.4 (4), we have $A[\neg A[\mathrm{T}]], \neg\neg A[\mathrm{T}] \vdash A[\mathrm{F}]$. Combining them, we have $A[\neg A[\mathrm{T}]] \vdash A[\mathrm{F}]$.

From the right-hand side to the left-hand side. The claim follows by letting X be $A[\mathrm{T}]$ in Lemma 4.4 (4).

(2) We have $\neg\neg A[\neg A[\mathrm{T}]] \dashv\vdash \neg\neg A[\mathrm{T}], \neg\neg A[\mathrm{F}]$ from (1). By Lemma 4.3 (9), we have $\neg\neg A[\neg A[\mathrm{T}]] \dashv\vdash \neg\neg A[\mathrm{T}] \wedge \neg\neg A[\mathrm{F}]$. Hence $\neg A[\neg A[\mathrm{T}]] \dashv\vdash \neg(\neg\neg A[\mathrm{T}] \wedge \neg\neg A[\mathrm{F}])$ holds. The right-hand side is equivalent to $\neg A[T] \vee \neg A[F]$ by Lemma 4.7 and Lemma 4.3 (8).

(3) The direction from the right-hand side to the left-hand side immediately follows from (2).

The direction from the left-hand side to the right-hand side. By Lemma 4.4 (1), we have $\neg A, X \vdash \neg A[\mathrm{T}]$. Hence we get $\neg A, X \vdash \neg A[\mathrm{T}] \vee \neg A[\mathrm{F}]$. By Lemma 4.4 (2) and (3), we similarly have $\neg A, \neg X \vdash \neg A[\mathrm{T}] \vee \neg A[\mathrm{F}]$. Hence we have $X \vee \neg X, \neg A \vdash \neg A[\mathrm{T}] \vee \neg A[\mathrm{F}]$. By Lemma 4.3 (6), we have $\neg A \vdash \neg A[\mathrm{T}] \vee \neg A[\mathrm{F}]$. So this direction holds. □

Remark. (1) The claim (3) does not hold in the minimal logic with implication and existential quantification. A counterexample is $A := \neg(X \to \bot \to Y)$. Then the left-hand side holds by taking $\bot \to Y$ as X, but the right-hand side is equivalent to $((\bot \to Y) \to \bot) \to \bot$, which is Peirce's formula in minimal logic.

(2) We could not directly prove the claim (3) without the claim (2).

(3) In order to eliminate universal quantification in a similar way, we can directly show the corresponding claim $\forall X\neg A \dashv\vdash \neg A[T], \neg A[F]$ in $\mathrm{L}\to\exists\bot$ extended with universal quantification.

As a byproduct of Proposition 5.1, we have the next theorem, which says that $\exists X\neg A$ implies that we can choose the witness $\neg A[\mathrm{T}]$ as X.

Theorem 5.2 (Second-Order Witness Theorem) $\exists X\neg A \dashv\vdash \neg A[X := \neg A[X := \mathrm{T}]]$ *holds in both* $\mathrm{L}\neg\wedge\exists$ *and* $\mathrm{L}\to\exists\bot$.

Proof. It is proved by Proposition 5.1 (2) and (3). □

5.1 Existential Quantification Elimination in $\mathrm{L}\neg\wedge\exists$

In this subsection, we will study $\mathrm{L}\neg\wedge\exists$.

For a formula A, we define $\widetilde{A}$, which does not contain $\exists$ and is equivalent to A in a negated context.

Definition 5.3 Given a formula A, the formula $\widetilde{A}$ is defined by

$$\begin{aligned}
&\widetilde{X} = X,\\
&\widetilde{\bot} = \bot,\\
&\widetilde{(\neg A)} = \neg\widetilde{A},\\
&\widetilde{(A \wedge B)} = \widetilde{A} \wedge \widetilde{B},\\
&\widetilde{(\exists X A)} = \widetilde{A}[X := \mathrm{T}] \vee \widetilde{A}[X := \mathrm{F}].
\end{aligned}$$

The next lemma shows that A and $\widetilde{A}$ are equivalent in a negated context.

Lemma 5.4 $\neg\neg A \dashv\vdash \neg\neg\widetilde{A}$ *in* $\mathrm{L}\neg\wedge\exists$.

Proof. By induction on A. We will show only interesting cases.

Case $A_1 \wedge A_2$ follows from Lemma 4.3 (4) and induction hypothesis.

Case $\exists X A$. $\neg\neg\exists X A$ is equivalent to $\neg\neg\exists X\neg\neg A$ by Lemma 4.3 (5), which is equivalent to $\neg\neg\exists X\neg\neg\widetilde{A}$ by induction hypothesis, which is equivalent to $\neg\neg(\neg\neg\widetilde{A}[\mathrm{T}] \vee \neg\neg\widetilde{A}[\mathrm{F}])$ by Proposition 5.1 (3), which is equivalent to $\neg\neg(\widetilde{A}[\mathrm{T}] \vee \widetilde{A}[\mathrm{F}])$ by Lemma 4.3 (7). Hence $\neg\neg\exists X A \dashv\vdash \neg\neg\widetilde{(\exists X A)}$ holds. □

The next lemma is another key property. It says that any formula is equivalent to a conjunctive normal form.

Lemma 5.5 *In* $\mathrm{L}\neg\wedge\exists$, *for a given formula* A, *there are variables* $X_1, \ldots, X_n$ $(n \geq 0)$ *and a formula* B *such that* $A \dashv\vdash X_1 \wedge \ldots \wedge X_n \wedge \neg B$.

Proof. By induction on A. We will give $\vec{X}$ and B such that $A \dashv\vdash \vec{X} \wedge \neg B$.

Case X. Let $\vec{X}$ be X and B be F.

Case $\bot$. Let $\vec{X}$ be empty and B be T.

Case $\neg A_1$. Let $\vec{X}$ be empty and B be A_1.

Case $A_1 \wedge A_2$. By induction hypothesis, there are $\vec{X}_1, B_1, \vec{X}_2, B_2$ such that $A_1 \dashv\vdash \vec{X}_1 \wedge \neg B_1$ and $A_2 \dashv\vdash \vec{X}_2 \wedge \neg B_2$. Let $\vec{X}$ be $\vec{X}_1 \wedge \vec{X}_2$ and B be $B_1 \vee B_2$.

Case $\exists X A_1$. By induction hypothesis, there are $X_1, \ldots, X_n$ and B_1 such that $A_1 \dashv\vdash X_1 \wedge \ldots \wedge X_n \wedge \neg B_1$.

Case 1 when $X \in \{X_1, \ldots, X_n\}$. We can suppose $X = X_1$ and $X \neq X_i \quad (1 < i)$. Let $\vec{X}$ be $X_2 \wedge \ldots \wedge X_n$ and B be $B_1[X_1 := \mathrm{T}]$. By Lemma 4.4 (1), we have $X_1 \wedge \ldots \wedge X_n \wedge \neg B_1 \dashv\vdash X_1 \wedge \ldots \wedge X_n \wedge \neg B_1[X_1 := \mathrm{T}]$. Hence we have $\exists X_1 A_1 \vdash X_2 \wedge \ldots \wedge X_n \wedge \neg B_1[X_1 := \mathrm{T}]$. We also have $X_2 \wedge \ldots \wedge X_n \wedge \neg B_1[X_1 := \mathrm{T}] \vdash \exists X_1 A_1$ since $\mathrm{T} \wedge X_2 \wedge \ldots \wedge X_n \wedge \neg B_1[X_1 := \mathrm{T}] \vdash \exists X_1 A_1$. Hence we have $\exists X_1 A_1 \dashv\vdash \vec{X} \wedge \neg B$.

Case 2 when $X \notin \{X_1, \ldots, X_n\}$. Let $\vec{X}$ be $X_1 \wedge \ldots \wedge X_n$ and B be $\neg(\neg B_1[\mathrm{T}] \vee \neg B_1[\mathrm{F}])$. Then we have $\exists X A_1 \dashv\vdash X_1 \wedge \ldots \wedge X_n \wedge \exists X \neg B_1$. By Proposition 5.1 (3) and Lemma 4.3 (8), we have $\exists X A_1 \dashv\vdash \vec{X} \wedge \neg B$. □

Recently Sakagawa and Kashima independently showed the following claim by using sequent calculus. We can prove it easily from Lemma 5.5.

Proposition 5.6 $\vdash_J A$ *if and only if* $\vdash_K A$ *for* $J = \mathrm{L}\neg\wedge\exists\bot$.

Proof. It is sufficient to show the right-hand side implies the left-hand side. By Lemma 5.5, we have $X_1, \ldots, X_n, B$ such that $A \dashv\vdash X_1 \wedge \ldots \wedge X_n \wedge \neg B$. Hence we have $\vdash_K X_1 \wedge \ldots \wedge X_n \wedge \neg B$. Therefore $X_1, \ldots, X_n$ are empty and A is $\neg B$. By Proposition 4.8, we have $\vdash_J A$. $\Box$

By using these lemmas, we can prove the existential quantification elimination in $\mathrm{L}\neg \wedge \exists$.

Proof of Theorem 3.1. For a given formula A, by Lemma 5.5, we have $\vec{X}$ and C such that $A \dashv\vdash \vec{X} \wedge \neg C$. By Lemma 5.4, we have $\neg C \dashv\vdash \neg\widetilde{C}$. Let B be $\vec{X} \wedge \neg\widetilde{C}$. Then B does not contain $\exists$ and $A \dashv\vdash B$ holds. $\Box$

5.2 Existential Quantification Elimination in $\mathrm{L} \to \exists\bot$

In this subsection, we will study $\mathrm{L} \to \exists\bot$.

For a formula A with only existential quantification followed by negation, we define $\widetilde{A}$, which does not contain $\exists$ and is equivalent to A.

Definition 5.7 Given an $\exists\neg$-formula A, the formula $\widetilde{A}$ is defined by

$$\begin{aligned}
&\widetilde{X} = X, \\
&\widetilde{\bot} = \bot, \\
&\widetilde{(A \to B)} = \widetilde{A} \to \widetilde{B}, \\
&\widetilde{(\exists X \neg A)} = \neg\widetilde{A}[X := \mathrm{T}] \vee \neg\widetilde{A}[X := \mathrm{F}].
\end{aligned}$$

The next proposition shows the equivalence between A and $\widetilde{A}$.

Proposition 5.8 *$A \dashv\vdash \widetilde{A}$ holds for any $\exists\neg$-formula A in* $\mathrm{L} \to \exists$.

Proof. By induction on A.

Cases X and $\bot$. The claim trivially holds.

Case $A \to B$. By induction hypothesis.

Case $\exists X \neg A$. $\exists X \neg A$ is equivalent to $\exists X \neg\widetilde{A}$ by induction hypothesis, which is equivalent to $\neg\widetilde{A}[\mathrm{T}] \vee \neg\widetilde{A}[\mathrm{F}]$ by Proposition 5.1 (3). $\Box$

Now we can prove the existential quantification elimination in $\mathrm{L} \to \exists\bot$.

Proof of Theorem 3.5. The claim follows immediately from Proposition 5.8. $\Box$

6 Decidability Proof

We will prove all the decidability theorems given in Sections 2 and 3.

We will use proof normalization in our proof, which is given by the next theorem.

Theorem 6.1 (Strong Normalization) *Any proof is strongly normalizing in both systems* $\mathrm{L}\neg \wedge \exists$ *and* $\mathrm{L}\neg \wedge \exists\bot$.

This theorem is proved in a similar way to strong normalization in the second-order logic with $\to, \wedge, \vee, \forall, \exists$ proved in [8].

We will use propositional systems corresponding to our systems.

Definition 6.2 The logical system L$\neg\wedge$ is defined as the logical system obtained from L$\neg\wedge\exists$ by deleting $\exists$. The logical system L$\neg\wedge\bot$ is also defined from L$\neg\wedge\exists\bot$ by deleting $\exists$. $J-\exists$ denotes L$\neg\wedge$ and L$\neg\wedge\bot$ respectively when J is L$\neg\wedge\exists$ and L$\neg\wedge\exists\bot$.

Remark. $J-\exists$ is a fragment of intuitionistic propositional logic and minimal propositional logic.

Theorem 6.3 *Provability in* L$\neg\wedge$ *and* L$\neg\wedge\bot$ *is decidable.*

This claim immediately follows from decidability of provability in intuitionistic propositional logic and minimal propositional logic.

Theorem 6.4 *Let J be one of* L$\neg\wedge\exists$ *and* L$\neg\wedge\exists\bot$. *For given formulas $B_1,\ldots,B_n,A$, there are formulas $B'_1,\ldots,B'_n,A'$ without $\exists$ such that $B_1,\ldots,B_n\vdash A$ is provable in J if and only if $B'_1,\ldots,B'_n\vdash A'$ is provable in $J-\exists$.*

Proof. By Theorem 3.1, formulas $B_1,\ldots,B_n,A$ are equivalent to some formulas $B'_1,\ldots,B'_n,A'$ without $\exists$ in J respectively. Let Γ be $B_1,\ldots,B_n$ and Γ' be $B'_1,\ldots,B'_n$.

From the left-hand side to the right-hand side. Suppose $\Gamma\vdash A$ is provable in J. Then $\Gamma'\vdash A'$ is provable in J. By Theorem 6.1, $\Gamma'\vdash A'$ has a normal proof. By the subformula property, this proof does not contain $\exists$. Hence $\Gamma'\vdash A'$ is provable in $J-\exists$.

The right-hand side trivially implies the left-hand side. $\Box$

We can finish our proof of the decidability theorems for L$\neg\wedge\exists$ and L$\neg\wedge\exists\bot$ given in Section 3.

Proof of Theorems 3.2 and 3.3. The claims are proved by combining Theorem 6.4 and Theorem 6.3. $\Box$

Finally we prove the decidability theorems for $\lambda\neg\wedge\exists$ and $\lambda\neg\wedge\exists\bot$.

Proof of Theorems 2.1 and 2.2. These claims immediately follow from Theorems 3.2 and 3.3. $\Box$

7 Concluding Remarks

We have investigated existential quantification elimination and showed inhabitance in the type systems with existential types is decidable by reducing it to provability in propositional systems without existential types.

Future work would be extending our existential quantification elimination to (1) a type system with arrow and existential types, and (2) type checking and type inference for existential types.

We would like to solve inhabitance problem in a system with arrow and existential types. As we remarked in Section 5, our existential quantification elimination cannot directly apply to that system. However, our elimination theorem will give a new insight into that question.

Decidability of type checking and type inference for existential types is another interesting question. We hope our existential quantification elimination will give lights for that problem.

Acknowledgments

We would like to thank Professor Stefano Berardi for comments. We would also like to thank Professor Yukiyoshi Kameyama and Professor Sachio Hirokawa for discussions.

References

1. G. Dowek and Y. Jiang, Eigenvariables, bracketing and the decidability of positive minimal predicate logic, *Theoretical Computer Science* **360** (1–3) (2006) 193–208.
2. K. Fujita, Galois Embedding from Polymorphic Types into Existential Types, In: Proceedings of 7th International Conference on Typed Lambda Calculi and Applications (TLCA 2005), *Lecture Notes in Computer Science* 3461 (2005) 194–208.
3. M.H. Löb, Embedding First Order Predicate Logic in Fragments of Intuitionistic Logic, *Journal of Symbolic Logic* **41** (4) (1976) 705–718.
4. R. Matthes, Non-strictly positive fixed-points for classical natural deduction, *Annals of Pure and Applied Logic* **133** (1–3) (2005) 205–230.
5. Y. Minamide, G. Morrisett, and R. Harper, Typed closure conversion, In: *Proceeding of 23rd ACM Symposium on Principles of Programming Languages (POPL'96)* (1996) 271–283.
6. J.C. Mitchell and G.D. Plotkin, Abstract types have existential type, *ACM Transactions on Programming Languages and Systems* **10** (3) (1988) 470–502.
7. M. Tatsuta and G. Mints, A simple proof of second-order strong normalization with permutative conversions, *Annals of Pure and Applied Logic* **136** (1–2) (2005) 134–155.
8. M. Tatsuta, Simple saturated sets for disjunction and second-order existential quantification, In: Proceedings of 8th International Conference on Typed Lambda Calculi and Applications (TLCA 2007), *Lecture Notes in Computer Science* 4583 (2007) 366–380.
9. A.S. Troelstra, Metamathematical Investigation of Intuitionistic Arithmetic and Analysis, *Lecture Notes in Mathematics* **344** (1973).
10. A.S. Troelstra and H. Schwichtenberg, *Basic Proof Theory, Second Edition* (Cambridge Univ Press, 2000).
11. P. Urzyczyn, Inhabitation in typed lambda-calculi, In: Proceedings of 3rd International Conference on Typed Lambda Calculi and Applications (TLCA 1997), *Lecture Notes in Computer Science* 1210 (1997) 373–389.
12. J. B. Wells, Typability and type checking in System F are equivalent and undecidable, *Annals of Pure and Applied Logic* **98** (1–3) (1999) 111–156.

Appdendix

A Proof of Lemma 4.6

Proof. (1) By induction on A. Use Lemma 4.3 (2) and (4) with induction hypothesis. Case $A \to B$ is proved by Lemma 4.3 (3) in $\mathrm{L} \to \exists\bot$.

(2) By induction on $\mathcal{C}$. When $\mathcal{C}$ is the hole, the claim follows from (1). The other cases are proved by induction hypothesis.

(3) Let Γ be $B_1, \ldots, B_n$ and Γ^- be $B_1^-, \ldots, B_n^-$. By induction on the proof. Cases are considered according to the last rule.

Case $(\exists I)$. By induction hypothesis, we have $(A[X := B])^-$. By (2), we have $A^-[B^-]$. Hence $\exists X A^-$ holds. Therefore we have $\neg\neg\exists X A^-$, which is $(\exists X A)^-$.

Case $(\exists E)$. By induction hypothesis, we have $\Gamma^-, A^- \vdash C^-$. Hence $\Gamma^-, \exists X A^- \vdash C^-$ holds. Therefore we have $\Gamma^-, \neg\neg\exists X A^- \vdash \neg\neg C^-$. By (1), we have $\Gamma^-, (\exists X A)^- \vdash C^-$.

Case $(\bot C)$. By induction hypothesis, we have $\Gamma^-, \neg A^- \vdash \bot$. Hence $\Gamma^- \vdash \neg\neg A^-$ holds. By (1), we have $\Gamma^- \vdash A^-$.

The other cases are proved straightforwardly by induction hypothesis.

(4) By induction on A. Case $\neg A$ is proved by induction hypothesis. Case $A \wedge B$ is proved by induction hypothesis and Lemma 4.3 (4). Case $A \to B$ is proved by induction hypothesis and Lemma 4.3 (3). Case $\exists X A$ is proved by induction hypothesis and Lemma 4.3 (5). □

B Proof of Lemma 4.7

Proof. By Lemma 4.6 (3), we have $A^- \dashv\vdash B^-$ in J. In J, $\neg\mathcal{C}[A]$ is equivalent to $\neg(\mathcal{C}[A])^-$ by Lemma 4.6 (4), which is equivalent to $\neg\mathcal{C}^-[A^-]$ by Lemma 4.6 (2), which is equivalent to $\neg\mathcal{C}^-[B^-]$, which is equivalent to $\neg\mathcal{C}[B]$ by Lemma 4.6 (2) and (4). □