HAL
archives-ouvertes.fr

A Finite First-Order Presentation of Set Theory

Stéphane Vaillant

▶ **To cite this version:**

Stéphane Vaillant. A Finite First-Order Presentation of Set Theory. RR-4344, INRIA. 2001. inria-00072244

HAL Id: inria-00072244
https://hal.inria.fr/inria-00072244

Submitted on 23 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A Finite First-Order Presentation of Set Theory

Stéphane Vaillant

N° 4344

Décembre 2001

THÈME 2

ISSN 0249-6399 ISRN INRIA/RR--4344--FR+ENG

INRIA
ROCQUENCOURT

A Finite First-Order Presentation of Set Theory

Stéphane Vaillant

Thème 2 — Génie logiciel
et calcul symbolique
Projet Logical

Rapport de recherche n° 4344 — Décembre 2001 — 35 pages

Abstract: We present a first-order formalization of set theory which has a finite number of axioms. Its syntax is similar to that often used in textbooks: it provides an encoding of the comprehension symbol. We prove that this formalization is a "conservative extension" of Zermelo's set theory. In fact the proof is more general and applies to other variants of Zermelo's set theory like ZF.

This formalization rests upon an encoding of the comprehension binder in a language of explicit substitution. This presentation of set theory is also described as a *deduction modulo* system and the proof of equivalence is done within this formalism.

Key-words: set theory, explicit substitution, deduction modulo, first-order predicate logic

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

Une présentation finie de la théorie des ensembles au premier ordre

Résumé : Dans cet article on donne une présentation finie de la theorie des ensembles au premier ordre dont la syntaxe est similaire à celle souvent rencontrée dans la litérature : elle fournit un codage du symbole de compréhension. On prouve que cette présentation est une "extension conservative" de la théorie de Zermelo. En fait la preuve est plus générale et s'applique à d'autres variantes de la théorie de Zermelo comme ZF par exemple.

Cette formalisation est fondée sur un codage du symbole lieur qu'est le symbole de compréhension dans un langage de substitution explicite.

Cette présentation est aussi décrite comme une *théorie modulo*, la preuve de conservativité se faisant dans ce formalisme.

Mots-clés : théorie des ensembles, substitution explicite, déduction modulo, logique des prédicats du premier ordre

Contents

Introduction

Computer based theorem proving in set theory requires a finite presentation of its axioms. For instance [BLM+86] used the von Neumann, Bernays and Gödel axiomatization of set theory [Men87] in the framework of first-order resolution. This axiomatization is expressed in first-order predicate logic and has a finite number of axioms; the device used in this formalization is *proposition as class* encoding using only few axioms for class construction.

Having a finite number of axioms is essential but having notations for set constructions is also important. For instance, in textbooks Z (or ZF) is given with existential axioms and immediately is introduced the comprehension symbol which is a notation for the construction defined by the comprehension scheme. Extended propositions built with this notation are in fact abbreviations for primitive propositions. A language where this symbol is built-in can be defined but it is not a first-order language anymore. Such a presentation is given in [Dow95] and proved to be equivalent to Zermelo's theory. We call it Z_b.

Reusing HOL-$\lambda\sigma$ [DHK99], we give a finite first-order presentation of Zermelo's set theory providing an encoding of such a symbol. The principle is as follows: propositions of Z_b are encoded as first-order terms using de Bruijn indices to encode occurrences of bound variables and using an explicit substitution. This gives an almost natural encoding as it preserves the structure of the propositions. These terms are *propositional contents* and we need to express them as propositions; this is why a predicate ε is introduced such that, for instance, $\varepsilon(\text{“}A \wedge B\text{”}) \text{ is equivalent to } \varepsilon(\text{“}A\text{”}) \wedge \varepsilon(\text{“}B\text{”})$. So, on one hand this device allows us to have a first-order formalization and on the other hand having the ability to quantify over these terms allows us to express the comprehension scheme as one axiom.

Having given this presentation of set theory, we prove that it is what we can call a "conservative extension" of Z_b. The precise meaning is: there exists a translation $P \mapsto P'$ such that for any proposition P of Z_b, P is provable in Z_b if and only if P' is provable in the new presentation.

In order to do the proof we rely on two important points: the structure of propositions (with a binder) is preserved by the translation and, since the theory associated with the encoding is decidable, we express it as a congruence on propositions and we work in *deduction modulo* [DHK98].

This presentation in *deduction modulo* deserves its own interest, in particular for proof search, since a cut-elimination theorem has been proved.

Preliminaries

Let $\mathcal{F}$ be a set of function symbols with arity. Let $\mathcal{X}$ be an infinite set of variables. $\mathcal{T}(\mathcal{F},\mathcal{X})$ denotes the first-order algebra generated from $\mathcal{F}$ and $\mathcal{X}$.

Rewrite systems A rewrite system is given by a first-order term algebra $\mathcal{T}(\mathcal{F})$ and by a set $\mathcal{R}$ composed of pairs of terms from $\mathcal{T}(\mathcal{F},\mathcal{X})$, each pair being written $l \to r$, where all variables of r have at least one occurrence in l.

A rule $l \to r$ in which any variable has at most one occurrence in l is said to be left linear.

Let $(\mathcal{T}(\mathcal{F}),\mathcal{R})$ be a rewrite system, $\to_{\mathcal{R}}$ denotes the one-step rewrite relation generated from $\mathcal{R}$. We write $\to^+_{\mathcal{R}}$ for its transitive closure, $\to^*_{\mathcal{R}}$ for its reflexive transitive closure and $\leftrightarrow^*_{\mathcal{R}}$ for its reflexive symmetric transitive closure.

We call a rewrite relation $\to$ confluent if for all a,b,c, $a \to^* b \wedge a \to^* c$ implies $\exists d\ b \to^* d \wedge c \to^* d$, locally confluent if for all a,b,c, $a \to b \wedge a \to c$ implies $\exists d\ b \to^* d \wedge c \to^* d$, and strongly confluent if for all a,b,c, $a \to b \wedge a \to c$ implies $\exists d\ b \to d \wedge c \to d$.

Let t be a term, t is said to be in normal form if there does not exist any term t' such that $t \to t'$. We call a rewrite relation $\to$ strongly normalizing if all derivations are finite, and weakly normalizing if all terms have a normal form.

Two rewrite relations $\to_{\mathcal{R}}$ and $\to_{\mathcal{S}}$ are said to commute if for all a,b,c, $a \to^*_{\mathcal{R}} b \wedge a \to^*_{\mathcal{S}} c$ implies $\exists d\ b \to^*_{\mathcal{S}} d \wedge c \to^*_{\mathcal{R}} d$, and to strongly commute if for all a,b,c, $a \to_{\mathcal{R}} b \wedge a \to_{\mathcal{S}} c$ implies $\exists d\ b \to_{\mathcal{S}} d \wedge c \to_{\mathcal{R}} d$.

First-order predicate logic Let $\mathcal{X}$, $\mathcal{F}$ and $\mathcal{P}$ be the sets of variables, function symbols and predicate symbols, $\mathcal{T}(\mathcal{P},\mathcal{F},\mathcal{X})$ denotes the language of propositions (formulas).

Let P be a proposition, $FV(P)$ denotes the set of its free variables and $BV(P)$ the set of its bound variables.

A valuation σ is a function from $\mathcal{X}$ into $\mathcal{T}(\mathcal{F},\mathcal{X})$ such that $\sigma(x) = x$ except for a finite number of variables. This function may be extended to a function from $\mathcal{T}(\mathcal{F},\mathcal{X})$ into $\mathcal{T}(\mathcal{F},\mathcal{X})$ in two ways: either as a grafting denoted $\{x_1 \mapsto t_1; \ldots; x_n \mapsto t_n\}$ or as a substitution (with capture avoiding) denoted $\{t_1/x_1; \ldots; t_n/x_n\}$.

Let σ be a valuation and $\{t_i/x_i\}$ the associated substitution (again named σ). $Dom(\sigma)$ denotes the set of the variables x such that $\sigma(x) \neq x$; $Ran(\sigma)$ denotes the set of the variables of the terms t_i (such that $t_i \neq x_i$) and $Var(\sigma)$ denotes $Dom(\sigma) \cup Ran(\sigma)$.

Expressions are considered modulo α-conversion: we can always assume that when a substitution σ is applied to a term whose head symbol is a binder, the bound variable, x, does not occur in the substitution (that is x does not belong to $Var(\sigma)$).

A proposition P is said to be provable if the sequent $\vdash P$ is provable. Given a set $\mathcal{A}$ of sentences (closed propositions), P is said to be provable under $\mathcal{A}$ if there exists a finite list A of propositions of $\mathcal{A}$ such that the sequent $A \vdash P$ is provable.

If P is a proposition then $\bar{\forall}P$ denotes its universal closure.

If P is a proposition and t a term then $\{t/x\}P$ is said to be an instance of $\forall x\, P$. If P, Q and R are propositions such that Q is an instance of P and R is an instance of Q then R is an instance of P.

Let L and L' be first order languages such that L is a subset of L'. Let T be a theory of L and T' a theory of L'. T' is said to be a *conservative extension* of T if for any proposition P of L, $\vdash_T P$ holds if and only if $\vdash_{T'} P$ holds.

New definition Let L and L' be first order languages. Let F be a mapping from L to L'. Let T be a theory of L and T' a theory of L'. T' is said to *conservatively interpret* T if for any proposition P of L, $\vdash_T P$ if and only if $\vdash_{T'} F(P)$.

1 Zermelo's set theory with a binder: Z_b

In [Dow95], two presentations of Zermelo's set theory [Kri98, Sup72] are given. One with existence axioms and the other, which we call Z_b, with a notation for set constructed by comprehension: the comprehension symbol which is a binder. These presentations are equivalent in the sense that the later is a conservative extension of the former and the later can be encoded in the former (this proof requiring the presence of the extensionality axiom).

Here we give these two presentations and, in the sequel, we shall show that the presentation given in the following section conservatively interprets Z_b.

Set theory with existence axioms

It is expressed in first-order predicate logic. The language is given by the following inductive definition:

$$\begin{array}{lllll} \text{propositions} & P & ::= & P \Rightarrow P \mid P \wedge P \mid P \vee P \mid \neg P \mid \bot \mid \forall x\, P \mid \exists x\, P \\ & & & \mid t = t \mid t \in t \\ \text{terms} & t & ::= & x \end{array}$$

where x is a variable from an infinite set $\mathcal{X}$ of variables.

Substitution is defined as usual.

There are five axioms and two axiom schemes:

- (Pairing axiom) $\forall x \forall y \exists A \forall z (z \in A \Leftrightarrow (z = x \vee z = y))$
- (Powerset axiom) $\forall x \exists A \forall y (y \in A \Leftrightarrow \forall z (z \in y \Rightarrow z \in x))$
- (Sum axiom) $\forall x \exists A \forall y (y \in A \Leftrightarrow \exists z (y \in z \wedge z \in x))$
- (Comprehension scheme)

  for any proposition P whose free variables are among $z, x_1 \ldots x_n$,
  $\forall x_1 \ldots \forall x_n \forall y \exists A \forall z (z \in A \Leftrightarrow (z \in y \wedge P))$
- (Extensionality axiom) $\forall x \forall y ((\forall z (z \in x \Leftrightarrow z \in y)) \Rightarrow x = y)$
- (Equality axiom) $\forall x\ x = x$
- (Equality scheme)

  for any proposition P whose free variables are among $z, x_1 \ldots x_n$,
  $\forall x_1 \ldots \forall x_n \forall x \forall y (x = y \Rightarrow (\{x/z\}P \Rightarrow \{y/z\}P))$

Set theory with a binder: Z_b

The language is given by the following inductive definition:

$$\begin{array}{lllll} \text{propositions} & P & ::= & P \Rightarrow P \mid P \wedge P \mid P \vee P \mid \neg P \mid \bot \mid \forall x\, P \mid \exists x\, P \\ & & & \mid t = t \mid t \in t \\ \text{terms} & t & ::= & x \mid \{t, t\} \mid \mathcal{P}(t) \mid \mathcal{U}(t) \mid \{x \in t \mid P\} \end{array}$$

where x is a variable from an infinite set $\mathcal{X}$ of variables.

As the term language uses a binder, the definition of substitution needs to be extended. Using the fact that we consider terms/propositions modulo α-conversion, the substitution is defined by:

$$\begin{array}{l} \{t/x\}y = t \quad \text{if } x = y \\ \{t/x\}y = y \quad \text{if } x \neq y \\ \{t/x\}(\{t_1, t_2\}) = \{\{t/x\}t_1, \{t/x\}t_2\} \\ \{t/x\}(\square t_1) = \square\{t/x\}t_1 \quad \text{where } \square \text{ is in } \{\mathcal{P}, \mathcal{U}\} \\ \{t/x\}(\{y \in t' \mid P\}) = \{y \in (\{t/x\}t') \mid \{t/x\}P\} \\ \{t/x\}(t_1 \square t_2) = \{t/x\}t_1 \square \{t/x\}t_2 \quad \text{where } \square \text{ is in } \{=, \in\} \\ \{t/x\}\bot = \bot \\ \{t/x\}(\neg P) = \neg(\{t/x\}P) \\ \{t/x\}(P \square Q) = \{t/x\}P \square \{t/x\}Q \quad \text{where } \square \text{ is in } \{\Rightarrow, \wedge, \vee\} \\ \{t/x\}(\square y\, P) = \square y\, (\{t/x\}P) \quad \text{where } \square \text{ is in } \{\forall, \exists\} \end{array}$$

There are five axioms and two axiom schemes:

- (Pairing axiom) $\forall x \forall y \forall z (z \in \{x, y\} \Leftrightarrow (z = x \vee z = y))$
- (Powerset axiom) $\forall x \forall y (y \in \mathcal{P}(x) \Leftrightarrow \forall z (z \in y \Rightarrow z \in x))$
- (Sum axiom) $\forall x \forall y (y \in \mathcal{U}(x) \Leftrightarrow \exists z (y \in z \wedge z \in x))$
- (Axiom scheme of comprehension)

  for any proposition P whose free variables are among $z, x_1 \ldots x_n$,
  $\forall x_1 \ldots \forall x_n \forall y \forall z \, (z \in \{z \in y \mid P\} \Leftrightarrow (z \in y \wedge P))$
- (Extensionality axiom) $\forall x \forall y ((\forall z (z \in x \Leftrightarrow z \in y)) \Rightarrow x = y)$
- (Equality axiom) $\forall x \; x = x$
- (Equality scheme)

  for any proposition P whose free variables are among $z, x_1 \ldots x_n$
  $\forall x_1 \ldots \forall x_n \forall x \forall y (x = y \Rightarrow (\{x/z\}P \Rightarrow \{y/z\}P))$

Proof theory, for instance sequent calculus, is the same as in first-order predicate logic with the exception of the new definition of the substitution.

Remark: In usual presentations of Zermelo's theory there is no equality scheme but only a finite number of axioms expressing that equality is a congruence on terms and propositions. Nonetheless, using either these axioms or the scheme gives equivalent presentations.

2 A finite first-order presentation of Z_b

Here we present a theory in first-order predicate logic with a finite number of axioms. In the rest of the paper, this theory will be proved to conservatively interpret Z_b: for any proposition P of Z_b there exists a proposition P' in this theory such that P is provable in Z_b if and only if P' is provable in this theory.

We have adapted the work that has been done in [DHK99] for higher order logic to Z_b. This presentation can be seen as a syntactic transformation of Z_b: first we get rid of the binder by coding bound variables by de Bruijn indices, then the language of term/proposition of Z_b is seen as a two-sorted first-order algebra. The substitution of Z_b is then simulated by an explicit substitution. Therefore we add another sort *subst*, axioms to deal with the explicit substitution and axioms to deal with the encoding of the propositions of Z_b.

We work in many-sorted first-order predicate logic with equality.

- Sorts: *set*, *o*, *subst*
- Function symbols:

$\{\_,\_\}$	rank	$(set,set)set$	$\mathtt{1}$	sort	set
$\mathcal{P}(\_)$	rank	$(set)set$	$\_[\![\_]\!]$	rank	$(o,subst)o$
$\mathcal{U}(\_)$	rank	$(set)set$	$\_[\_]$	rank	$(set,subst)set$
$\{\_ \mid \_\}$	rank	$(set,o)set$	$\mathtt{id}$	sort	$subst$
$\_\doteq\_,\ \_\dot\in\_$	rank	$(set,set)o$	$\_\cdot\_$	rank	$(set,subst)subst$
$\_\dot\Rightarrow\_$	rank	$(o,o)o$	$\uparrow$	sort	$subst$
$\_\dot\wedge\_,\ \_\dot\vee\_$	rank	$(o,o)o$	$\Uparrow(\_)$	rank	$(subst)subst$
$\dot\neg(\_)$	rank	$(o)o$	$\_\circ\_$	rank	$(subst,subst)subst$
$\dot\bot$	sort	o			
$\dot\forall(\_),\ \dot\exists(\_)$	rank	$(o)o$			

- Predicate symbols: ε of rank (o) and an equality symbol for each sort.
- Axioms of equality.
- Axioms for the explicit substitution:

$$
\begin{array}{l}
\forall a \forall s \forall t\ (a[s])[t] = a[s \circ t] \\
\forall a \forall s \forall t\ (a[\![s]\!])[\![t]\!] = a[\![s \circ t]\!] \\
\forall a \forall s\ \mathtt{1}[a \cdot s] = a \\
\forall s\ \mathtt{1}[\Uparrow(s)] = \mathtt{1} \\
\forall s \forall t\ \mathtt{1}[\Uparrow(s) \circ t] = \mathtt{1}[t] \\
\forall s \forall t \forall u\ (s \circ t) \circ u = s \circ (t \circ u) \\
\forall a \forall s \forall t\ (a \cdot s) \circ t = a[t] \cdot (s \circ t) \\
\forall a \forall s \uparrow \circ (a \cdot s) = s \\
\forall s \uparrow \circ \Uparrow(s) = s \circ \uparrow \\
\forall s \forall t \uparrow \circ (\Uparrow(s) \circ t) = s \circ (\uparrow \circ t) \\
\forall s \forall t \Uparrow(s) \circ \Uparrow(t) = \Uparrow(s \circ t) \\
\forall s \forall t \forall u \Uparrow(s) \circ (\Uparrow(t) \circ u) = \Uparrow(s \circ t) \circ u \\
\forall a \forall s \forall t \Uparrow(s) \circ (a \cdot t) = a \cdot (s \circ t) \\
\forall s\ \mathtt{id} \circ s = s \\
\forall s\ s \circ \mathtt{id} = s \\
\Uparrow(\mathtt{id}) = \mathtt{id} \\
\forall a\ a[\mathtt{id}] = a \\
\forall a\ a[\![\mathtt{id}]\!] = a
\end{array}
$$

$$\forall x \forall y \forall s\ \{x, y\}[s] = \{x[s], y[s]\}$$
$$\forall x \forall s\ \mathcal{P}(x)[s] = \mathcal{P}(x[s])$$
$$\forall x \forall s\ \mathcal{U}(x)[s] = \mathcal{U}(x[s])$$
$$\forall x \forall Z \forall s\ \{x \mid Z\}[s] = \{x[s] \mid Z[\![\Uparrow(s)]\!]\}$$
$$\forall a \forall b \forall s\ (a \doteq b)[\![s]\!] = a[s] \doteq b[s]$$
$$\forall a \forall b \forall s\ (a \dot{\in} b)[\![s]\!] = a[s] \dot{\in} b[s]$$
$$\forall X \forall Y \forall s\ (X \dot{\Rightarrow} Y)[\![s]\!] = X[\![s]\!] \dot{\Rightarrow} Y[\![s]\!]$$
$$\forall X \forall Y \forall s\ (X \dot{\wedge} Y)[\![s]\!] = X[\![s]\!] \dot{\wedge} Y[\![s]\!]$$
$$\forall X \forall Y \forall s\ (X \dot{\vee} Y)[\![s]\!] = X[\![s]\!] \dot{\vee} Y[\![s]\!]$$
$$\forall X \forall Y \forall s\ (\dot{\neg} X)[\![s]\!] = \dot{\neg}(X[\![s]\!])$$
$$\forall X \forall s\ (\dot{\forall} X)[\![s]\!] = \dot{\forall}(X[\![\Uparrow(s)]\!])$$
$$\forall X \forall s\ (\dot{\exists} X)[\![s]\!] = \dot{\exists}(X[\![\Uparrow(s)]\!])$$
$$\forall s\ \dot{\bot}[\![s]\!] = \dot{\bot}$$

- Axioms for the encoding:

$$\forall X \forall Y\ \varepsilon(X \dot{\wedge} Y) \Leftrightarrow (\varepsilon(X) \wedge \varepsilon(Y))$$
$$\forall X \forall Y\ \varepsilon(X \dot{\vee} Y) \Leftrightarrow (\varepsilon(X) \vee \varepsilon(Y))$$
$$\forall X \forall Y\ \varepsilon(X \dot{\Rightarrow} Y) \Leftrightarrow (\varepsilon(X) \Rightarrow \varepsilon(Y))$$
$$\forall X\ \varepsilon(\dot{\neg} X) \Leftrightarrow \neg\varepsilon(X)$$
$$\varepsilon(\dot{\bot}) \Leftrightarrow \bot$$
$$\forall X\ \varepsilon(\dot{\forall} X) \Leftrightarrow \forall y\, \varepsilon(X[\![y \cdot \mathtt{id}]\!])$$
$$\forall X\ \varepsilon(\dot{\exists} X) \Leftrightarrow \exists y\, \varepsilon(X[\![y \cdot \mathtt{id}]\!])$$

- Axioms for set theory:

$$\forall x \forall y \forall z(\varepsilon(z \dot{\in} \{x, y\}) \Leftrightarrow (\varepsilon(z \doteq x) \vee \varepsilon(z \doteq y)))$$

$$\forall x \forall y(\varepsilon(y \dot{\in} \mathcal{P}(x)) \Leftrightarrow \forall z(\varepsilon(z \dot{\in} y) \Rightarrow \varepsilon(z \dot{\in} x)))$$

$$\forall x \forall y(\varepsilon(y \dot{\in} \mathcal{U}(x)) \Leftrightarrow \exists z(\varepsilon(y \dot{\in} z) \wedge \varepsilon(z \dot{\in} x)))$$

$$\forall p \forall y \forall z(\varepsilon(z \dot{\in} \{y \mid p\}) \Leftrightarrow (\varepsilon(z \dot{\in} y) \wedge \varepsilon(p[\![z \cdot \mathtt{id}]\!])))$$

$$\forall x \forall y((\forall z(\varepsilon(z \dot{\in} x) \Leftrightarrow \varepsilon(z \dot{\in} y))) \Rightarrow \varepsilon(x \doteq y))$$

$$\forall x\ \varepsilon(x \doteq x)$$

$$\forall p \forall x \forall y(\varepsilon(x \doteq y) \Rightarrow (\varepsilon(p[\![x \cdot \mathtt{id}]\!]) \Rightarrow \varepsilon(p[\![y \cdot \mathtt{id}]\!])))$$

As we shall prove in the following section the union of the theories dealing with the explicit substitution and the encoding is decidable, therefore it is reasonable to avoid explicit use of these axioms in proofs. This is why we give an intermediate presentation of set theory in *deduction modulo*.

3 A finite presentation of Z_b in deduction modulo: Z_{es}

In this section we give a presentation of set theory, called Z_{es}, which is not first-order but which is a *theory modulo*. A theory modulo can be seen as a theory expressed in a first-order language where propositions are identified modulo a congruence.

Its language is the same as the one of the preceding presentation except that it has no equality predicate and that there is an infinite number of indices. The substitution is defined as in many-sorted first-order predicate logic.

The system is composed of two theories: one expressed as a decidable congruence (the terminating and confluent rewrite system $\sigma_{\in} \cup \mathcal{L}$) which defines the explicit substitution and the encoding; and the other is a finite set of axioms for set theory (it is the same as the one given in the preceding section). The proof system is sequent calculus modulo this congruence.

In the next section we shall prove that Z_{es} conservatively interprets Z_b and then state that the finite presentation given in the preceding section is a conservative extension of Z_{es}. In order to do that we need to prove properties of the rewrite system of Z_{es} and cut-elimination in sequent calculus modulo the congruence generated by this rewrite system.

We are now giving the formal definition of the system Z_{es}.

The language, rewrite system and axioms of Z_{es}

The system is split in two components. The first is an axiom free theory modulo, called $\in_{es}$, which is not specific to set theory except for it's language. Then we add axioms for set theory and we obtain the theory modulo called Z_{es}.

The term language is given by the following ranked signature in many-sorted first-order predicate logic with the three sorts *set*, *o* and *subst*.

$\{\_,\_\}$	rank	$(set,set)set$
$\mathcal{P}(\_)$	rank	$(set)set$
$\mathcal{U}(\_)$	rank	$(set)set$
$\{\_ \mid \_\}$	rank	$(set,o)set$
$\_\dot{=}\_$, $\_\dot{\in}\_$	rank	$(set,set)o$
$\_\dot{\Rightarrow}\_$	rank	$(o,o)o$
$\_\dot{\wedge}\_$, $\_\dot{\vee}\_$	rank	$(o,o)o$
$\dot{\neg}(\_)$	rank	$(o)o$
$\dot{\bot}$	sort	o
$\dot{\forall}(\_)$, $\dot{\exists}(\_)$	rank	$(o)o$

`n`	sort	*set*
$\_[\![\_]\!]$	rank	$(o,subst)o$
$\_[\_]$	rank	$(set,subst)set$
`id`	sort	*subst*
$\_\cdot\_$	rank	$(set,subst)subst$
$\uparrow$	sort	*subst*
$\Uparrow(\_)$	rank	$(subst)subst$
$\_\circ\_$	rank	$(subst,subst)subst$

Remark: `n` ranges over positive natural numbers. This introduces an infinite number of symbols but as remarked in [CHL96] concerning the definition of $\lambda\sigma_{\Uparrow}$ each index `n+1` can be encoded as the term $1[\uparrow^n]$ in a language with the only index `1` such as was done in the preceding section.

There is only one predicate symbol: ε of rank (o).

The theory $\in_{es}$ is expressed as a congruence defined by three sets of rewrite rules: σ, $\in_\sigma$ and $\mathcal{L}$. σ is the calculus of explicit substitution that comes from $\lambda\sigma_{\Uparrow}$ [CHL96], $\in_\sigma$ are the rules that express the interaction of the substitution with the symbols of the term language we have defined and finally $\mathcal{L}$ gives the meaning of terms of sort o.

- Rules of σ

clos	$(a[s])[t]$	$\to$	$a[s \circ t]$
clos'	$(a[\![s]\!])[\![t]\!]$	$\to$	$a[\![s \circ t]\!]$
vs1	$\mathtt{n}[\uparrow]$	$\to$	$\mathtt{n+1}$
vs2	$\mathtt{n}[\uparrow \circ s]$	$\to$	$\mathtt{n+1}\,[s]$
fvc	$\mathtt{1}[a \cdot s]$	$\to$	a
fvl1	$\mathtt{1}[\Uparrow(s)]$	$\to$	$\mathtt{1}$
fvl2	$\mathtt{1}[\Uparrow(s) \circ t]$	$\to$	$\mathtt{1}[t]$
rvc	$\mathtt{n+1}[a \cdot s]$	$\to$	$\mathtt{n}[s]$
rvl1	$\mathtt{n+1}[\Uparrow(s)]$	$\to$	$\mathtt{n}[s \circ \uparrow]$
rvl2	$\mathtt{n+1}[\Uparrow(s) \circ t]$	$\to$	$\mathtt{n}[s \circ (\uparrow \circ t)]$
ae	$(s \circ t) \circ u$	$\to$	$s \circ (t \circ u)$

me	$(a \cdot s) \circ t$	$\to$	$a[t] \cdot (s \circ t)$
sc	$\uparrow \circ (a \cdot s)$	$\to$	s
sl1	$\uparrow \circ \Uparrow(s)$	$\to$	$s \circ \uparrow$
sl2	$\uparrow \circ (\Uparrow(s) \circ t)$	$\to$	$s \circ (\uparrow \circ t)$
l1	$\Uparrow(s) \circ \Uparrow(t)$	$\to$	$\Uparrow(s \circ t)$
l2	$\Uparrow(s) \circ (\Uparrow(t) \circ u)$	$\to$	$\Uparrow(s \circ t) \circ u$
le	$\Uparrow(s) \circ (a \cdot t)$	$\to$	$a \cdot (s \circ t)$
idl	$\mathtt{id} \circ s$	$\to$	s
idr	$s \circ \mathtt{id}$	$\to$	s
lid	$\Uparrow(\mathtt{id})$	$\to$	$\mathtt{id}$
id	$a[\mathtt{id}]$	$\to$	a
id'	$a[\![\mathtt{id}]\!]$	$\to$	a

Remark: since a substitution can be applied to terms of sort *set* and terms of sort o we have added another substitution application symbol ($[\![]\!]$) and the corresponding rewrite rules (clos' and id').

- Rules of $\in_\sigma$

$\{x, y\}[s]$	$\to$	$\{x[s], y[s]\}$
$\mathcal{P}(x)[s]$	$\to$	$\mathcal{P}(x[s])$
$\mathcal{U}(x)[s]$	$\to$	$\mathcal{U}(x[s])$
$\{x \mid Y\}[s]$	$\to$	$\{x[s] \mid Y[\![\Uparrow(s)]\!]\}$
$(x \doteq y)[\![s]\!]$	$\to$	$x[s] \doteq y[s]$
$(x \dot\in y)[\![s]\!]$	$\to$	$x[s] \dot\in y[s]$

$(X \dot\Rightarrow Y)[\![s]\!]$	$\to$	$X[\![s]\!] \dot\Rightarrow Y[\![s]\!]$
$(X \dot\wedge Y)[\![s]\!]$	$\to$	$X[\![s]\!] \dot\wedge Y[\![s]\!]$
$(X \dot\vee Y)[\![s]\!]$	$\to$	$X[\![s]\!] \dot\vee Y[\![s]\!]$
$(\dot\neg X)[\![s]\!]$	$\to$	$\dot\neg(X[\![s]\!])$
$(\dot\forall X)[\![s]\!]$	$\to$	$\dot\forall(X[\![\Uparrow(s)]\!])$
$(\dot\exists X)[\![s]\!]$	$\to$	$\dot\exists(X[\![\Uparrow(s)]\!])$
$\dot\bot[\![s]\!]$	$\to$	$\dot\bot$

- Let $\sigma_\in$ be $\sigma \cup \in_\sigma$
- Rules of $\mathcal{L}$

$\varepsilon(X \dot\Rightarrow Y)$	$\to$	$\varepsilon(X) \Rightarrow \varepsilon(Y)$	
$\varepsilon(X \dot\wedge Y)$	$\to$	$\varepsilon(X) \wedge \varepsilon(Y)$	
$\varepsilon(X \dot\vee Y)$	$\to$	$\varepsilon(X) \vee \varepsilon(Y)$	
$\varepsilon(\dot\neg X)$	$\to$	$\neg\varepsilon(X)$	
$\varepsilon(\dot\bot)$	$\to$	$\bot$	
$\varepsilon(\dot\forall X)$	$\to$	$\forall y\, \varepsilon(X[\![y \cdot \mathtt{id}]\!])$	with $y \notin \mathrm{FV}(X)$
$\varepsilon(\dot\exists X)$	$\to$	$\exists y\, \varepsilon(X[\![y \cdot \mathtt{id}]\!])$	with $y \notin \mathrm{FV}(X)$

The theory Z_{es} is the theory $\in_{es}$ plus the following set of axioms (it is the same as given in the preceding section).

$$\forall x \forall y \forall z (\varepsilon(z \dot\in \{x, y\}) \Leftrightarrow (\varepsilon(z \doteq x) \vee \varepsilon(z \doteq y)))$$

$$\forall x \forall y (\varepsilon(y \dot\in \mathcal{P}(x)) \Leftrightarrow \forall z (\varepsilon(z \dot\in y) \Rightarrow \varepsilon(z \dot\in x)))$$

$$\forall x \forall y (\varepsilon(y \dot\in \mathcal{U}(x)) \Leftrightarrow \exists z (\varepsilon(y \dot\in z) \wedge \varepsilon(z \dot\in x)))$$

$$\forall p \forall y \forall z (\varepsilon(z \dot\in \{y \mid p\}) \Leftrightarrow (\varepsilon(z \dot\in y) \wedge \varepsilon(p[\![z \cdot \mathtt{id}]\!])))$$

$$\forall x \forall y ((\forall z (\varepsilon(z \dot\in x) \Leftrightarrow \varepsilon(z \dot\in y))) \Rightarrow \varepsilon(x \doteq y))$$

$$\forall x\ \varepsilon(x \doteq x)$$

$$\forall p \forall x \forall y (\varepsilon(x \doteq y) \Rightarrow (\varepsilon(p[\![x \cdot \mathtt{id}]\!]) \Rightarrow \varepsilon(p[\![y \cdot \mathtt{id}]\!])))$$

Sequent calculus modulo: the proof system of $\in_{es}/\mathrm{Z}_{es}$

Sequent calculus modulo [DHK98] is given in fig.1. This calculus is first-order sequent calculus in which propositions are matched modulo a congruence.

In this paper the congruence for the proof system of Z_{es} is the congruence generated by the rewrite system $\sigma_{\in} \cup \mathcal{L}$.

For example the sequent $\varepsilon(\dot{\forall}(1 \doteq 1)) \vdash \forall x\, \varepsilon(x \doteq x)$ is provable using the rule *axiom* since the two propositions are congruent: $\varepsilon(\dot{\forall}(1 \doteq 1)) \to_{\mathcal{L}} \forall x\, \varepsilon((1 \doteq 1)[\![x \cdot id]\!]) \to^*_{\sigma_{\in}} \forall x\, \varepsilon(x \doteq x)$.

$$\frac{\Gamma, Q_1, Q_2 \vdash \Delta}{\Gamma, P \vdash \Delta}\ \text{contr-l} \quad \text{if } P \equiv Q_1 \equiv Q_2$$

$$\frac{\Gamma \vdash Q_1, Q_2, \Delta}{\Gamma \vdash P, \Delta}\ \text{contr-r} \quad \text{if } P \equiv Q_1 \equiv Q_2$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, P \vdash \Delta}\ \text{weak-l} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash P, \Delta}\ \text{weak-r}$$

$$\frac{\Gamma, P \vdash \Delta \quad \Gamma \vdash Q, \Delta}{\Gamma \vdash \Delta}\ \text{cut} \quad \text{if } P \equiv Q$$

$$\frac{}{P \vdash Q}\ \text{axiom} \quad \text{if } P \equiv Q$$

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, C \vdash \Delta}\ \Rightarrow\text{-l} \quad \text{if } C \equiv (A \Rightarrow B)$$

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash C, \Delta}\ \Rightarrow\text{-r} \quad \text{if } C \equiv (A \Rightarrow B)$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, C \vdash \Delta}\ \wedge\text{-l} \quad \text{if } C \equiv (A \wedge B) \qquad \frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash C, \Delta}\ \wedge\text{-r} \quad \text{if } C \equiv (A \wedge B)$$

$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, C \vdash \Delta}\ \vee\text{-l} \quad \text{if } C \equiv (A \vee B) \qquad \frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash C, \Delta}\ \vee\text{-r} \quad \text{if } C \equiv (A \vee B)$$

$$\frac{\Gamma \vdash A, \Delta}{\Gamma, C \vdash \Delta}\ \neg\text{-l} \quad \text{if } C \equiv \neg A \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash C, \Delta}\ \neg\text{-r} \quad \text{if } C \equiv \neg A$$

$$\frac{}{\Gamma, A \vdash \Delta}\ \bot\text{-l} \quad \text{if } A \equiv \bot$$

$$\frac{\Gamma, \{t/x\}P \vdash \Delta}{\Gamma, Q \vdash \Delta}\ (x, P, t)\ \forall\text{-l} \quad \text{if } Q \equiv (\forall x\, P)$$

$$\frac{\Gamma, P \vdash \Delta}{\Gamma, Q \vdash \Delta}\ (x, P)\ \exists\text{-l} \quad \text{if } Q \equiv (\exists x\, P) \text{ and } x \notin \mathrm{FV}(\Gamma\Delta)$$

$$\frac{\Gamma \vdash P, \Delta}{\Gamma \vdash Q, \Delta}\ (x, P)\ \forall\text{-r} \quad \text{if } Q \equiv (\forall x\, P) \text{ and } x \notin \mathrm{FV}(\Gamma\Delta)$$

$$\frac{\Gamma \vdash \{t/x\}P, \Delta}{\Gamma \vdash Q, \Delta}\ (x, P, t)\ \exists\text{-r} \quad \text{if } Q \equiv (\exists x\, P)$$

Figure 1: Rules of sequent calculus modulo a congruence $\equiv$.

- Language:

$$\begin{array}{lcll} \text{terms} & a & ::= & x_t \mid \mathtt{n} \mid a[s] \\ \text{substitutions} & s & ::= & x_s \mid \mathtt{id} \mid \uparrow \mid \Uparrow(s) \mid a \cdot s \mid s \circ s \end{array}$$

where x_t and x_s are elements of two infinite sets of variables of sort *term* and *subst* respectively.

- Rules:

$$\begin{array}{lrcl} \text{clos} & (a[s])[t] & \to & a[s \circ t] \\ \text{vs1} & \mathtt{n}[\uparrow] & \to & \mathtt{n{+}1} \\ \text{vs2} & \mathtt{n}[\uparrow \circ s] & \to & \mathtt{n{+}1}\,[s] \\ \text{fvc} & \mathtt{1}[a \cdot s] & \to & a \\ \text{fvl1} & \mathtt{1}[\Uparrow(s)] & \to & \mathtt{1} \\ \text{fvl2} & \mathtt{1}[\Uparrow(s) \circ t] & \to & \mathtt{1}[t] \\ \text{rvc} & \mathtt{n{+}1}[a \cdot s] & \to & \mathtt{n}[s] \\ \text{rvl1} & \mathtt{n{+}1}[\Uparrow(s)] & \to & \mathtt{n}[s \circ \uparrow] \\ \text{rvl2} & \mathtt{n{+}1}[\Uparrow(s) \circ t] & \to & \mathtt{n}[s \circ (\uparrow \circ t)] \\ \text{ae} & (s \circ t) \circ u & \to & s \circ (t \circ u) \\ \text{me} & (a \cdot s) \circ t & \to & a[t] \cdot (s \circ t) \\ \text{sc} & \uparrow \circ (a \cdot s) & \to & s \\ \text{sl1} & \uparrow \circ \Uparrow(s) & \to & s \circ \uparrow \\ \text{sl2} & \uparrow \circ (\Uparrow(s) \circ t) & \to & s \circ (\uparrow \circ t) \\ \text{l1} & \Uparrow(s) \circ \Uparrow(t) & \to & \Uparrow(s \circ t) \\ \text{l2} & \Uparrow(s) \circ (\Uparrow(t) \circ u) & \to & \Uparrow(s \circ t) \circ u \\ \text{le} & \Uparrow(s) \circ (a \cdot t) & \to & a \cdot (s \circ t) \\ \text{idl} & \mathtt{id} \circ s & \to & s \\ \text{idr} & s \circ \mathtt{id} & \to & s \\ \text{lid} & \Uparrow(\mathtt{id}) & \to & \mathtt{id} \\ \text{id} & a[\mathtt{id}] & \to & a \end{array}$$

- Notations:

$$\uparrow^n \text{ is defined by } \begin{cases} \uparrow^1 & ::= \uparrow \\ \uparrow^{n+1} & ::= \uparrow \circ (\uparrow^n) \quad \text{for } n \geq 1 \end{cases}$$

$$\Uparrow^n(s) \text{ is defined by } \begin{cases} \Uparrow^0(s) & ::= s \\ \Uparrow^{n+1}(s) & ::= \Uparrow(\Uparrow^n(s)) \quad \text{for } n \geq 0 \end{cases}$$

Figure 2: The $\sigma_{\Uparrow}$-calculus.

3.1 Properties of the rewrite system $\sigma_{\in} \cup \mathcal{L}$

We are going to prove that the rewrite system $\sigma_{\in} \cup \mathcal{L}$ is strongly normalizing and confluent using the fact that $\sigma_{\in} \cup \mathcal{L}$ can be translated into an *explicit reduction system*.

3.1.1 Explicit reduction systems

Introduction The idea of *explicit reduction systems* (XRS) [Pag98] is to extend the notion of $\lambda\sigma_{\Uparrow}$-calculus [CHL96] to any term signature with binding symbols. This allows to encode a higher-order rewrite system into a first-order one.

As an example the rule: $t \in \{x \in A \mid P\} \to t \in A \wedge \{t/x\}P$ constitutes such a higher-order system.

Another example of XRS is the $\lambda\sigma_{\Uparrow}$-calculus which can be seen as the substitution calculus $\sigma_{\Uparrow}$ (fig.2) the signature of which is extended with symbols for application and abstraction, and to which is added rules for the interaction of the explicit substitution with these symbols and a computational rule for β-reduction.

Crafting of an XRS There are two sorts: *term* and *subst*. A signature Γ on the sort *term* is given. The signature Γ extended with the signature of the $\sigma_{\Uparrow}$-calculus is written $\Gamma_{\Uparrow}$.

σ_Γ is the set of rules composed of $\sigma_{\Uparrow}$ and of a rule

$$f(a_1,\ldots,a_n)[s] \to_{f_{\Uparrow}} f(a_1[\Uparrow^{p_1}(s)],\ldots,a_n[\Uparrow^{p_n}(s)])$$

for every symbol f of Γ the arity of which is n; $(p_1,\ldots,p_n)$ being n given natural numbers called the binding arity of f. Binding arity indicates for each argument the binding height of the symbol for each of its arguments.

Let R be a set of rules on the signature $\Gamma_{\Uparrow}$. When the following conditions on R hold, the system $\sigma_\Gamma \cup R$ is an XRS:

1. All the rules are left linear.
2. Both members of each rule have the sort *term*.
3. Each left member is a term of the algebra generated by Γ.

In this paper we use $R = \emptyset$, but it may be used in future work to extend the congruence of $\in_{es}$ with rules like $t \dot\in \{A \mid P\} \to t \dot\in A \dot\wedge P[\![t \cdot \mathtt{id}]\!]$.

Proposition 3.1 *[Pag98] The relations $\to_{\sigma_{\Uparrow}}$ and $\to_{\sigma_\Gamma}$ are strongly normalizing and confluent.*

3.1.2 Properties of the rewrite system $\sigma_\in \cup \mathcal{L}$

Let's consider the signature of $\in_{es}$ in which the sorts *set* and *o* are merged into one sort *term* and in which the symbol $[\![]\!]$ has been removed. This signature is the signature of an XRS; let's name it $\Gamma_{\Uparrow}$, with Γ being composed of the symbols: $\{\_,\_\}$, $\mathcal{P}(\_)$, $\mathcal{U}(\_)$, $\{\_ \mid \_\}$, $\dot=$, $\dot\in$, $\dot\Rightarrow$, $\dot\wedge$, $\dot\vee$, $\dot\neg$, $\dot\forall$, $\dot\exists$, $\dot\bot$.

Then to each symbol of Γ we assign a binding arity. (), (0) or (0,0) according to the arity of the symbol, with the following exceptions:

symbol	binding arity
$\dot\forall$	(1)
$\dot\exists$	(1)
$\{ \mid \}$	(0,1)

This defines a rewrite relation $\to_{\sigma_\Gamma}$ which, according to prop. 3.1, is strongly normalizing and confluent.

Let ϕ be the homomorphism from the term algebra of $\in_{es}$ into the term algebra of $\Gamma_{\Uparrow}$ such that each symbol is mapped onto itself except for $[\![]\!]$ that is mapped onto $[]$.

Lemma 3.2 *Let a and b be two terms of $\in_{es}$ of the same sort. 1) If $\phi(a) = \phi(b)$ then $a = b$. 2) If $a \to_{\sigma_\in} b$ then $\phi(a) \to_{\sigma_\Gamma} \phi(b)$. 3) If $\phi(a) \to_{\sigma_\Gamma} b'$ then there exists b such that $a \to_{\sigma_\in} b$ and $\phi(b) = b'$.*

Proof. (1) is done by a simple induction on a. (2) and (3) are done by case analysis on the head rewrite step. □

Proposition 3.3 *$\sigma_\in$ is confluent and strongly normalizing.*

Proof. (1) $\sigma_\in$ is confluent. If $a \to^*_{\sigma_\in} b$ and $a \to^*_{\sigma_\in} c$ then according to lemma 3.2.2 $\phi(a) \to^*_{\sigma_\Gamma} \phi(b)$ and $\phi(a) \to^*_{\sigma_\Gamma} \phi(c)$. $\to_{\sigma_\Gamma}$ is confluent (prop. 3.1) so there exists d' such that $\phi(b) \to^*_{\sigma_\Gamma} d'$ and $\phi(c) \to^*_{\sigma_\Gamma} d'$. Using lemma 3.2.3, there exist d_1 and d_2 such that $b \to^*_{\sigma_\in} d_1$, $c \to^*_{\sigma_\in} d_2$ and $\phi(d_1) = d' = \phi(d_2)$. We then conclude using lemma 3.2.1: $d_1 = d_2$.

(2) $\sigma_\in$ is strongly normalizing. Using lemma 3.2.2 we deduce that for any derivation $a \to_{\sigma_\in} \ldots \to_{\sigma_\in} b$ there exists a derivation $\phi(a) \to_{\sigma_\Gamma} \ldots \to_{\sigma_\Gamma} \phi(b)$ with the same length. We conclude using the fact that $\to_{\sigma_\Gamma}$ is strongly normalizing (prop. 3.1). □

Lemma 3.4 *The rewrite system $\sigma_\in \cup \mathcal{L}$ is strongly normalizing.*

Proof. We define a function f over propositions and terms of sort o.

$$
\begin{array}{l}
f(A \Rightarrow B) = f(A \wedge B) = f(A \vee B) = f(A) + f(B) \\
f(\neg A) = f(\forall x\, A) = f(\exists x\, A) = f(A) \\
f(\bot) = 0 \\
f(\varepsilon(t)) = f(t) \\
f(a \dot{\vee} b) = f(a \dot{\wedge} b) = f(a \dot{\Rightarrow} b) = f(a) + f(b) + 1 \\
f(\dot{\neg} a) = f(\dot{\forall} a) = f(\dot{\exists} a) = f(a) + 1 \\
f(\dot{\bot}) = 1 \\
f(a \doteq b) = f(a \dot{\in} b) = 0 \\
f(a[\![s]\!]) = f(a)
\end{array}
$$

1. We show, for every proposition t or term t of sort o, that if $t \to_{\mathcal{L}} t'$ then $f(t) > f(t')$ and if $t \to_{\sigma_\in} t'$ then $f(t) = f(t')$.

   The proof is done by induction on t: we give only the case where t is $\varepsilon(\dot{\forall} a)$ and $t \to_{\mathcal{L}} t'$. t' is then $\forall x\, \varepsilon(a[\![x \cdot \mathtt{id}]\!])$ and we check that $f(t) = 1 + f(a) > f(a) = f(t')$

2. Let's assume we have an infinite $\sigma_\in \cup \mathcal{L}$-reduction sequence; since $\to_{\sigma_\in}$ is strongly normalizing (prop. 3.3) this sequence can't contain an infinite $\sigma_\in$-reduction sequence and we construct an infinite sequence of terms (t_i) for which $t_i \to^*_{\sigma_\in} \to_{\mathcal{L}} t_{i+1}$. The sequence $(f(t_i))$ is then strictly decreasing due to (1).

□

Lemma 3.5 (Hindley-Rosen) *If $\mathcal{R}$ and $\mathcal{S}$ are two confluent relations over the same set and are commuting then $\mathcal{R} \cup \mathcal{S}$ is confluent [Hin64][Bar84, p.64].*

Lemma 3.6 *The rewrite system $\mathcal{L}$ is strongly confluent.*

Proof. It is an orthogonal combinatory rewrite system [KvOvR93]. □

Lemma 3.7 *The rewrite system $\sigma_\in \cup \mathcal{L}$ is confluent.*

Proof. See [DHK99]. First we prove that the relations $\to_{\mathcal{L}}$ and $\to^*_{\sigma_\in}$ are strongly commuting. The proof is done by case analysis on the $\mathcal{L}$-head rewrite step. We process only one case: $\varepsilon(\dot{\forall} t) \to_{\mathcal{L}} \forall x\, \varepsilon(t[\![x \cdot \mathtt{id}]\!])$ with $x \notin FV(t)$ and $\varepsilon(\dot{\forall} t) \to^*_{\sigma_\in} \varepsilon(\dot{\forall} t')$ with $t \to^*_{\sigma_\in} t'$. Since $x \notin FV(t)$, $x \notin FV(t')$ and then we have that $\varepsilon(\dot{\forall} t') \to_{\mathcal{L}} \forall x\, \varepsilon(t'[\![x \cdot \mathtt{id}]\!])$. We also have that $\forall x\, \varepsilon(t[\![x \cdot \mathtt{id}]\!]) \to^*_{\sigma_\in} \forall x\, \varepsilon(t'[\![x \cdot \mathtt{id}]\!])$.

Now, we are able to give the proof of the lemma. $\to_{\sigma_\in}$ is confluent (prop.3.3) and $\to_{\mathcal{L}}$ is strongly confluent; the relations $\to_{\mathcal{L}}$ and $\to^*_{\sigma_\in}$ are commuting so according to Hindley-Rosen lemma their union is confluent. □

3.2 Cut elimination in sequent calculus modulo $\sigma_\in \cup \mathcal{L}$

We use the results of [DW99]: in this article the method used to show cut elimination in deduction modulo a congruence ($\equiv$) is to represent proofs as terms and to show termination of a generalized β-reduction ($\triangleright$) on these terms; the strong normalization proof is based on the technique of reducibility candidates (Tait and Girard, [GLT89]).

By the mean of a *pre-model* to each proposition is associated a *reducibility candidate*. When the congruence $\equiv$ is *valid in the pre-model* then proofs normalize and so intuitionistic natural deduction (and intuitionistic sequent calculus) enjoys cut elimination.

Definition 3.8

1. *Proof terms are defined by:*

$$\begin{array}{rl} \pi ::= & \alpha \\ | & \lambda\alpha\,\pi \mid (\pi\ \pi') \\ | & (\pi,\pi') \mid fst(\pi) \mid snd(\pi) \\ | & i(\pi) \mid j(\pi) \mid (\delta\ \pi_1\ \alpha\pi_2\ \beta\pi_3) \\ | & (botelim\ \pi) \\ | & \lambda x\,\pi \mid (\pi\ t) \\ | & (t,\pi) \mid (exelim\ \pi\ x\alpha\pi') \end{array}$$

2. *Proof reduction rules are:*

$$\begin{array}{l} (\lambda\alpha\,\pi_1\ \pi_2) \rhd [\pi_2/\alpha]\pi_1 \\ fst(\pi_1,\pi_2) \rhd \pi_1 \\ snd(\pi_1,\pi_2) \rhd \pi_2 \\ (\delta\ i(\pi_1)\ \alpha\pi_2\ \beta\pi_3) \rhd [\pi_1/\alpha]\pi_2 \\ (\delta\ j(\pi_1)\ \alpha\pi_2\ \beta\pi_3) \rhd [\pi_1/\beta]\pi_3 \\ (\lambda x\,\pi\ t) \rhd [t/x]\pi \\ (exelim\ (t,\pi_1)\ x\alpha\pi_2) \rhd [t/x,\pi_1/\alpha]\pi_2 \\ (\delta\ \pi_1\ \alpha\pi_2\ \beta\pi_3) \rhd \pi_2 \\ (\delta\ \pi_1\ \alpha\pi_2\ \beta\pi_3) \rhd \pi_3 \\ (exelim\ \pi_1\ x\alpha\pi_2) \rhd \pi_2 \end{array}$$

3. *$\mathcal{SN}$ is the set of strongly normalizing proofs.*
4. *A proof term is said to be* neutral *if it is a proof variable or an elimination.*
5. *A set R of proof terms is a* reducibility candidate *if the three following conditions hold:*
   (a) *If $\pi \in R$ then $\pi \in \mathcal{SN}$.*
   (b) *If $\pi \in R$ and $\pi \rhd^* \pi'$ then $\pi' \in R$.*
   (c) *If π is neutral and if for every π' such that $\pi \rhd \pi'$, $\pi' \in R$ then $\pi \in R$.*
6. *$\mathcal{C}$ is the set of reducibility candidates.*

Definition 3.9 *A pre-model is:*

- *for each sort s a set $\mathcal{M}_s$.*
- *for each function symbol f of rank $(s_1,\ldots,s_n)s$ a function $\tilde{f}$ from $\mathcal{M}_{s_1} \times \ldots \times \mathcal{M}_{s_n}$ to $\mathcal{M}_s$.*
- *for each predicate P of rank $(s_1,\ldots,s_n)$ a function $\tilde{P}$ from $\mathcal{M}_{s_1} \times \ldots \times \mathcal{M}_{s_n}$ to $\mathcal{C}$.*

Definition 3.10 *Let A be a proposition and ϕ be a variable assignment. $|A|_\phi$ is defined by:*

- $|P(t_1,\ldots,t_n)|_\phi = \tilde{P}(|t_1|_\phi,\ldots,|t_n|_\phi)$
- $|A \Rightarrow B|_\phi = \tilde{\Rightarrow}(|A|_\phi,|B|_\phi)$ *with*
  $\tilde{\Rightarrow}(a,b) = \{\pi \in \mathcal{SN} \mid \pi \rhd^* \lambda\alpha\pi_1 \Rightarrow \forall\pi' \in a\ [\pi'/\alpha]\pi_1 \in b\}$
- $|A \wedge B|_\phi = \tilde{\wedge}(|A|_\phi,|B|_\phi)$ *with*
  $\tilde{\wedge}(a,b) = \{\pi \in \mathcal{SN} \mid \pi \rhd^* (\pi_1,\pi_2) \Rightarrow \pi_1 \in a \wedge \pi_2 \in b\}$
- $|A \vee B|_\phi = \tilde{\vee}(|A|_\phi,|B|_\phi)$ *with*
  $\tilde{\vee}(a,b) = \{\pi \in \mathcal{SN} \mid \pi \rhd^* i(\pi_1) \Rightarrow \pi_1 \in a \text{ or } \pi \rhd^* j(\pi_2) \Rightarrow \pi_2 \in b\}$
- $|\ \bot\ |_\phi = \mathcal{SN}$
- $|\forall_s x\ A|_\phi = \tilde{\forall}_{s,x,\phi}(\phi' \mapsto |A|_{\phi'})$ *with*
  $\tilde{\forall}_{s,x,\phi}(a) = \{\pi \in \mathcal{SN} \mid \pi \rhd^* \lambda x\pi_1 \Rightarrow \forall t{:}s\ \forall E \in \mathcal{M}_s\ [t/x]\pi_1 \in a(\phi + <x,E>)\}$

- $|\exists_s x\ A|_\phi = \tilde{\exists}_{s,x,\phi}(\phi' \mapsto |A|_{\phi'})$ *with*
  $\tilde{\exists}_{s,x,\phi}(a) = \{\pi \in \mathcal{SN} \mid \exists E \in \mathcal{M}_s\ \pi \triangleright^* (t, \pi_1) \Rightarrow \pi_1 \in a(\phi + <x, E>)\}$

Definition 3.11 *Let there be given a pre-model. A congruence $\equiv$ is valid in the pre-model if and only if for any propositions A and B, $A \equiv B$ implies, for any variable assignment ϕ, $|A|_\phi \equiv |B|_\phi$.*

Theorem 3.12 *Let there be given a pre-model. When $\equiv$ is valid in the pre-model then any proof in deduction modulo $\equiv$ is strongly normalizing.*

When the hypothesis of this theorem are verified, cut-elimination holds in intuitionistic natural deduction modulo and in intuitionistic sequent calculus modulo. To prove cut-elimination in classical sequent calculus modulo, we first have to define double negation of a proposition.

Definition 3.13

Double negation	**Light double negation**
$A' = \neg\neg A$ *if* A *is atomic*	$A'' = A$ *if* A *is atomic*
$(A \Rightarrow B)' = \neg\neg(A' \Rightarrow B')$	$(A \Rightarrow B)'' = A' \Rightarrow B'$
$(A \wedge B)' = \neg\neg(A' \wedge B')$	$(A \wedge B)'' = A' \wedge B'$
$(A \vee B)' = \neg\neg(A' \vee B')$	$(A \vee B)'' = A' \vee B'$
$\bot' = \neg\neg \bot$	$\bot'' = \bot$
$(\forall x\ A)' = \forall x \neg\neg A'$	$(\forall x\ A)'' = \forall x\ A'$
$(\exists x\ A)' = \exists x \neg\neg A'$	$(\exists x\ A)'' = \exists x\ A'$

Theorem 3.14 *[DW99] Let $\mathcal{E}$ be a set of equations on terms and $\mathcal{R}$ a set of rewrite rules on propositions. Let $\mathcal{R}'$ be the light double negation of $\mathcal{R}$: for each rule $l \to r$ of $\mathcal{R}$, $l \to r''$ is a rule of $\mathcal{R}'$.*

If the congruence $\equiv_{\mathcal{E}\mathcal{R}'}$ is valid in a pre-model then cut elimination holds in classical sequent calculus modulo $\equiv_{\mathcal{E}\mathcal{R}}$.

Application to $\in_{es}$

Given $\mathcal{L}'$, the double negation of $\mathcal{L}$, we give a pre-model in the system $\in_{es}$ and we prove that the congruence $\equiv_{\sigma \in \mathcal{L}'}$ (*i.e.* $\leftrightarrow^*_{\sigma \in \cup \mathcal{L}'}$) is valid in it. In particular, we have to relate the interpretations of the symbols $\forall$ and $\dot{\forall}$, the first symbol binding a variable of sort *set*; this is almost trivial since the meaning of terms of sort *set* is irrelevant.

Sorts are interpreted as the sets $\mathcal{M}_{set} = \{0\}$, $\mathcal{M}_{substt} = \{0\}$ and $\mathcal{M}_o = \mathcal{C}$.

The interpretation of symbols is given below:

$\{\_,\_\}$	$x, y \mapsto 0$
$\mathcal{P}(\_)$	$x \mapsto 0$
$\mathcal{U}(\_)$	$x \mapsto 0$
$\{\_ \mid \_\}$	$x, y \mapsto 0$
$\_ \doteq \_$	$x, y \mapsto \mathcal{SN}$
$\_ \dot{\in} \_$	$x, y \mapsto \mathcal{SN}$
$\mathtt{n}$	0
$\_[\![\_]\!]$	$x, s \mapsto x$
$\_[\_]$	$x, s \mapsto 0$
$\mathtt{id}$	0
$\_ \cdot \_$	$x, y \mapsto 0$
$\uparrow$	0
$\Uparrow(\_)$	$s \mapsto 0$
$\_ \circ \_$	$x, y \mapsto 0$
$\_ \dot{\Rightarrow} \_$	$\tilde{\dot{\Rightarrow}} : (a, b) \mapsto \tilde{\dot{\Rightarrow}}(\tilde{\dot{\neg}}(\tilde{\dot{\neg}}(a)), \tilde{\dot{\neg}}(\tilde{\dot{\neg}}(b)))$
$\_ \dot{\wedge} \_$	$\tilde{\dot{\wedge}} : (a, b) \mapsto \tilde{\dot{\wedge}}(\tilde{\dot{\neg}}(\tilde{\dot{\neg}}(a)), \tilde{\dot{\neg}}(\tilde{\dot{\neg}}(b)))$
$\_ \dot{\vee} \_$	$\tilde{\dot{\vee}} : (a, b) \mapsto \tilde{\dot{\vee}}(\tilde{\dot{\neg}}(\tilde{\dot{\neg}}(a)), \tilde{\dot{\neg}}(\tilde{\dot{\neg}}(b)))$
$\dot{\bot}$	$\tilde{\dot{\bot}} = \tilde{\bot}$
$\dot{\neg}(\_)$	$\tilde{\dot{\neg}} : a \mapsto \tilde{\dot{\Rightarrow}}(a, \tilde{\dot{\bot}})$
$\dot{\forall}(\_)$	$\tilde{\dot{\forall}} : a \mapsto \{\pi \in \mathcal{SN} \mid \pi \rhd^* \lambda x \cdot \pi_1 \Rightarrow \forall t{:}set\ \{t/x\}\pi_1 \in \tilde{\dot{\neg}}(\tilde{\dot{\neg}}(a))\}$
$\dot{\exists}(\_)$	$\tilde{\dot{\exists}} : a \mapsto \{\pi \in \mathcal{SN} \mid \pi \rhd^* (t, \pi_1) \Rightarrow \pi_1 \in \tilde{\dot{\neg}}(\tilde{\dot{\neg}}(a))\}$
$\varepsilon(\_)$	$x \mapsto x$

We then prove that for any propositions A and B, if $A \equiv_{\sigma_\in \mathcal{L}'} B$ then, for any assignment ϕ, $|A|_\phi = |B|_\phi$. We proceed by case analysis on the rule used in the head rewrite step:

- It's a rule of $\sigma_\in$; two cases are possible:

  1. The rule rewrites a term of sort *set* or *subst.*

     Since after rewriting the term has the same sort, the interpretation of the term is 0 before and after.

  2. The rule rewrites a term of sort o.

     If it rewrites $t[\![s_1]\!][\![s_2]\!]$ into $t[\![s_1 \circ s_2]\!]$ then, applying the definition, we check that $|t[\![s_1]\!][\![s_2]\!]|_\phi$ and $|t[\![s_1 \circ s_2]\!]|_\phi$ are equal to $|t|_\phi$.

     If this rule rewrites $(\dot{\forall}t)[\![s]\!]$ into $\dot{\forall}(t[\![\Uparrow(s)]\!])$ then we check that the interpretations of both terms are equal to $|t|_\phi$. The argument applies to all the other cases.

- It's a rule of $\mathcal{L}'$.

  If the rule is $\varepsilon(X \dot{\Rightarrow} Y) \to \neg\neg\varepsilon(X) \Rightarrow \neg\neg\varepsilon(Y)$ then A is $\varepsilon(A_1 \dot{\Rightarrow} A_2)$ and we have

  $$|\varepsilon(A_1 \dot{\Rightarrow} A_2)|_\phi = |A_1 \dot{\Rightarrow} A_2|_\phi = \tilde{\dot{\Rightarrow}}(|A_1|_\phi, |A_2|_\phi) = \tilde{\dot{\Rightarrow}}(\tilde{\dot{\neg}}(\tilde{\dot{\neg}}(|A_1|_\phi)), \tilde{\dot{\neg}}(\tilde{\dot{\neg}}(|A_2|_\phi)))$$

  with the last expression being equal to

  $$|\neg\neg\varepsilon(A_1) \Rightarrow \neg\neg\varepsilon(A_2)|_\phi \ .$$

  The argument is the same for the rules dealing with $\dot{\wedge}$, $\dot{\vee}$, $\dot{\neg}$ and $\dot{\bot}$.

  If the rule is $\varepsilon(\dot{\forall}X) \to \forall y\ \neg\neg\varepsilon(X[\![y \cdot \mathtt{id}]\!])$ (with $y \notin FV(A')$) then A is $\varepsilon(\dot{\forall}A')$ and B is $\forall y\ \neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])$. Then we have

  $$\begin{array}{rcl} |B|_\phi & = & |\forall y\ \neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])|_\phi \\ & = & \{\pi \in \mathcal{SN} \mid \pi \rhd^* \lambda y{\cdot}\pi_1 \Rightarrow \forall t{:}set\ \forall E{\in}\mathcal{M}_{set}\ [t/y]\pi_1 \in |\neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])|_{\phi + <y,E>}\} \end{array}$$

Moreover,

$$\begin{array}{rcl} |\neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])|_{\phi+<y,E>} & = & \tilde{\neg}(\tilde{\neg}(|A'[\![y \cdot \mathtt{id}]\!]|_{\phi+<y,E>})) \\ & = & \tilde{\neg}(\tilde{\neg}(|A'|_{\phi+<y,E>})) \\ & = & \tilde{\neg}(\tilde{\neg}(|A'|_{\phi})) \quad \text{since } y \notin FV(A') \end{array}$$

Since $\mathcal{M}_{set}$ is not empty and E is not used, we deduce

$$|B|_{\phi} \quad = \quad \{\pi \in \mathcal{SN} \mid \pi \rhd^* \lambda y{\cdot}\pi_1 \Rightarrow \forall t{:}set\ [t/y]\pi_1 \in \tilde{\neg}(\tilde{\neg}(|A'|_{\phi}))\}$$

This last expression is equal to $|\varepsilon(\dot{\forall}A')|_{\phi}$.

Finally, if the rule is $\varepsilon(\dot{\exists}X) \to \exists y\ \neg\neg\varepsilon(X[\![y \cdot \mathtt{id}]\!])$ (with $y \notin FV(A')$) then A is $\varepsilon(\dot{\exists}A')$, B is $\exists y\ \neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])$ and we follow the same argument:

$$\begin{array}{l} |B|_{\phi} \\ \quad = \quad |\exists y\ \neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])|_{\phi} \\ \quad = \quad \{\pi \in \mathcal{SN} \mid \exists E \in \mathcal{M}_{set}\ \pi \rhd^* (t, \pi_1) \Rightarrow \pi_1 \in |\neg\neg\varepsilon(A'[\![y \cdot \mathtt{id}]\!])|_{\phi+<y,E>}\} \\ \quad = \quad \{\pi \in \mathcal{SN} \mid \pi \rhd^* (t, \pi_1) \Rightarrow \pi_1 \in \tilde{\neg}(\tilde{\neg}(|A'|_{\phi}))\} \\ \quad = \quad |\varepsilon(\dot{\exists}A')|_{\phi} \end{array}$$

The hypotheses of theorem 3.14 hold, hence we can reformulate it for our framework:

Theorem 3.15 *In classical sequent calculus modulo the congruence $\leftrightarrow^*_{\sigma \in \cup \mathcal{L}}$ the* **cut** *rule is redundant.*

4 Z_{es} conservatively interprets Z_b

To prove that Z_{es} conservatively interprets Z_b we shall first use a translation from the language of Z_b into the language of $\in_{es}$ (*i.e.* the language of Z_{es}) such that a proposition is provable in Z_b if and only if its translation is provable in $\in_{es}$. The result does not depend on the particular axioms chosen in the theories: in fact it gives a way to encode any theory expressed in the language of Z_b into a theory in $\in_{es}$. The resulting theory has the same number of axioms as the source theory, therefore in the case of the theory Z_b we obtain a theory in $\in_{es}$ that has an infinite number of axioms. So a last result we are going to prove is that, in $\in_{es}$, the translation of each axiom scheme of Z_b is equivalent to a single axiom of Z_{es}.

We recall that the notion of provability used in Z_b is sequent calculus and in $\in_{es}$ it is sequent calculus modulo the congruence $\leftrightarrow^*_{\sigma_\in \cup \mathcal{L}}$.

4.1 The translation: F

Definition 4.1 *The function F (pre-cooking[DHK00]) takes a term t of Z_b, a list of variables and translates t to a term of $\in_{es}$.*

$$
\begin{array}{lcl}
F(x,l) & = & \begin{cases} x[\uparrow^{|l|}] & \text{if } x \notin l \\ \mathtt{i} & \text{if } i \text{ is the first occurrence of } x \text{ in the list } l \end{cases} \\
F(\{a,b\},l) & = & \{F(a,l), F(b,l)\} \\
F(\mathcal{U}(a),l) & = & \mathcal{U}(F(a,l)) \\
F(\mathcal{P}(a),l) & = & \mathcal{P}(F(a,l)) \\
F(\{x \in a \mid b\},l) & = & \{F(a,l) \mid F(b, x \cdot l)\} \\
F(a = b,l) & = & F(a,l) \doteq F(b,l) \\
F(a \in b,l) & = & F(a,l) \dot{\in} F(b,l) \\
F(P \Rightarrow Q,l) & = & F(P,l) \dot{\Rightarrow} F(Q,l) \\
F(P \vee Q,l) & = & F(P,l) \dot{\vee} F(Q,l) \\
F(P \wedge Q,l) & = & F(P,l) \dot{\wedge} F(Q,l) \\
F(\bot,l) & = & \dot{\bot} \\
F(\neg P,l) & = & \dot{\neg} F(P,l) \\
F(\forall x P,l) & = & \dot{\forall} F(P, x \cdot l) \\
F(\exists x P,l) & = & \dot{\exists} F(P, x \cdot l)
\end{array}
$$

We shall write $F(P)$ instead of $F(P,[])$. $F(P)$ is called the F-translation of P.

Example: the proposition $\forall x\ x = y$ of Z_b is translated as the term $\dot{\forall}(1 \doteq y[\uparrow])$ of sort o.

Properties of F
The lemmas/corollaries 4.2, 4.3, 4.4 and 4.5 are proved in [DHK00] in the case where F is a translation between two term algebras with de Bruijn indices. In the present case it is a translation from an algebra with named variables into an algebra with de Bruijn indices.

Lemma 4.2 *Let t be a term or a proposition of Z_b and $\vec{z}$,$\vec{y}$,$\vec{x}$ be lists of variables. If $FV(t) \cap \vec{y} = \emptyset$ then*

$$F(t, \vec{z}\vec{y}\vec{x}) \leftrightarrow^*_{\sigma_\in} F(t, \vec{z}\vec{x})[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})]$$

Proof. By induction on t.

- t is $\{v \in a \mid P\}$.

$$
\begin{array}{lll}
F(\{v \in a \mid P\}, \vec{z}\vec{y}\vec{x}) & & \\
\quad = & \{F(a, \vec{z}\vec{y}\vec{x}) \mid F(P, v\vec{z}\vec{y}\vec{x})\} & \\
\quad \leftrightarrow^*_{\sigma_\in} & \{F(a, \vec{z}\vec{x})[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \mid F(P, v\vec{z}\vec{x})[\Uparrow^{1+|\vec{z}|}(\uparrow^{|\vec{y}|})]\} & \text{(by induction hypothesis)} \\
\quad \leftarrow_{\sigma_\in} & \{F(a, \vec{z}\vec{x}) \mid F(P, v\vec{z}\vec{x})\}[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] & \\
\quad = & F(\{v \in a \mid P\}, \vec{z}\vec{x})[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] &
\end{array}
$$

  The argument is the same for the other function symbols and the logical connectors.

- t is a variable z_i occurring in $\vec{z}$ (z_i is the ith last added variable in $\vec{z}$).

$$\begin{array}{rcl} F(z_i, \vec{z}\vec{y}\vec{x}) & = & \mathtt{i} \\ & \leftrightarrow^*_{\sigma\in} & \mathtt{i}[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \text{ (since } i \leq |\vec{z}|) \\ & = & F(z_i, \vec{z}\vec{x})[\Uparrow^{|\vec{z}|+|\vec{x}|}(\uparrow^{|\vec{y}|})] \end{array}$$

- t is a variable y_i such that $y_i \notin \vec{z}$ and $y_i \in \vec{y}$: impossible.

- t is a variable x_i such that $x_i \notin \vec{z}\vec{y}$ and $x_i \in \vec{x}$.

$$\begin{array}{rcl} F(x_i, \vec{z}\vec{y}\vec{x}) & \leftrightarrow^*_{\sigma\in} & \mathtt{i}[\uparrow^{|\vec{y}|} \circ \uparrow^{|\vec{z}|}] \\ & \leftrightarrow^*_{\sigma\in} & \mathtt{i}[\uparrow^{|\vec{z}|} \circ \Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \\ & \leftrightarrow^*_{\sigma\in} & (\mathtt{i}+|\vec{z}|)[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \\ & = & F(x_i, \vec{z}\vec{x})[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \end{array}$$

- t is a variable $v \notin \vec{z}\vec{y}\vec{x}$.

$$\begin{array}{rcl} F(v, \vec{z}\vec{y}\vec{x}) & = & v[\uparrow^{|\vec{z}|+|\vec{y}|+|\vec{x}|}] \\ & \leftrightarrow^*_{\sigma\in} & v[\uparrow^{|\vec{x}|} \circ \uparrow^{|\vec{z}|} \circ \Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \\ & \leftrightarrow^*_{\sigma\in} & v[\uparrow^{|\vec{z}|+|\vec{x}|}][\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \\ & = & F(v, \vec{z}\vec{x})[\Uparrow^{|\vec{z}|}(\uparrow^{|\vec{y}|})] \end{array}$$

□

Lemma 4.3 (F is a homomorphism of substitutions)
Let t be a term and p be a proposition or a term of Z_b. If l is a list of variables all different from x and if $FV(t) \cap l = \emptyset$ then

$$F(\{t/x\}p, l) \leftrightarrow^*_{\sigma\in} \{F(t)/x\}F(p, l)$$

Proof. By induction on p.

- p is $\{v \in a \mid P\}$.

$$\begin{array}{rcll} & & F(\{t/x\}\{v \in a \mid P\}, l) & \\ & = & F(\{v \in \{t/x\}a \mid \{t/x\}P\}, l) & \\ & = & \{F(\{t/x\}a, l) \mid F(\{t/x\}P, vl)\} & \\ & \leftrightarrow^*_{\sigma\in} & \{\{F(t)/x\}F(a, l) \mid \{F(t)/x\}F(P, vl)\} & \text{(by induction hypothesis)} \\ & = & \{F(t)/x\}\{F(a, l) \mid F(P, vl)\} & \\ & = & \{F(t)/x\}F(\{v \in a \mid P\}, l) & \end{array}$$

  The argument is the same for the other function symbols and the logical connectors.

- p is a variable y. When $y \notin l$ and $y = x$, we have

$$F(\{t/x\}x, l) = F(t, l)$$

  and

$$\begin{array}{rcll} \{F(t)/x\}F(x, l) & = & \{F(t)/x\}\, x[\uparrow^{|l|}] & \text{since } x \notin l \\ & = & F(t)[\uparrow^{|l|}] & \end{array}$$

  $FV(t) \cap l = \emptyset$ so we apply lemma 4.2 with $\vec{x} = []$, $\vec{z} = []$ and $\vec{y} = l$, hence $F(t, l) \leftrightarrow^*_{\sigma\in} F(t)[\uparrow^{|l|}]$. When $y \notin l$ and $y \neq x$ it is obvious; so it is when $y \in l$ and $y \neq x$. The case where $y \in l$ and $y = x$ is impossible.

□

Lemma 4.4 (Substitution lemma) *Let a be a term or a proposition of Z_b, b be a term of Z_b, n and p be natural numbers such that $n \geq 1$ and $0 \leq p \leq n-1$. Let $x_n \ldots x_{n-p+1}$ be variables not occurring in b.*

$$\begin{array}{l} F(a, x_n \ldots x_{n-p+1}x_{n-p}x_{n-p-1} \ldots x_1)[\Uparrow^p(F(b, x_{n-p-1} \ldots x_1) \cdot \mathtt{id})] \\ \quad \leftrightarrow^*_{\sigma\in} F(\{b/x_{n-p}\}a, x_n \ldots x_{n-p+1}x_{n-p-1} \ldots x_1) \end{array}$$

Proof. By induction on a.

- a is $\{v \in t \mid P\}$.

$$\begin{array}{rl} & F(\{v \in t \mid P\}, x_n \dots x_1)[\Uparrow^p(F(b, x_{n-p-1} \dots x_1) \cdot \mathtt{id})] \\ = & \{F(t, x_n \dots x_1) \mid F(P, v x_n \dots x_1)\}[\Uparrow^p(F(b, x_{n-p-1} \dots x_1) \cdot \mathtt{id})] \\ \leftrightarrow^*_{\sigma\in} & \{F(t, x_n \dots x_1)[\Uparrow^p(F(b, x_{n-p-1} \dots x_1) \cdot \mathtt{id})] \mid \\ & \quad F(P, v x_n \dots x_1)[\Uparrow^{p+1}(F(b, x_{n-p-1} \dots x_1) \cdot \mathtt{id})]\} \end{array}$$

(v not occurring in b, we apply the induction hypothesis:)

$$\begin{array}{rl} \leftrightarrow^*_{\sigma\in} & \{F(\{b/x_{n-p}\}t, x_n \dots x_{n-p+1} x_{n-p-1} \dots x_1) \mid \\ & \quad F(\{b/x_{n-p}\}P, v x_n \dots x_{n-p+1} x_{n-p-1} \dots x_1)\} \\ = & F(\{b/x_{n-p}\}\{v \in t \mid P\}, x_n \dots x_{n-p+1} x_{n-p-1} \dots x_1) \end{array}$$

The argument is the same for the other symbols.

- a is x_{n-p}. When p is 0 the result is trivial so, we assume $p \geq 1$.

$F(a, x_n \dots x_1)[\Uparrow^p(F(b, x_{n-p-1} \dots x_1) \cdot \mathtt{id})] \leftrightarrow^*_{\sigma\in} F(b x_{n-p-1} \dots x_1)[\uparrow^p]$

$F(\{b/x_{n-p}\}a, x_n \dots x_{n-p+1} x_{n-p-1} \dots x_1) = F(b, x_n \dots x_{n-p+1} x_{n-p-1} \dots x_1)$

By hypothesis none of the variables $x_n \dots x_{n-p+1}$ occur in b, so, according to lemma 4.2, we have

$$F(b, x_n \dots x_{n-p+1} x_{n-p-1} \dots x_1) \leftrightarrow^*_{\sigma\in} F(b, x_{n-p-1} \dots x_1)[\uparrow^p]$$

- a is x_{n-i} with $i < p$. Both sides are convertible to `i+1`.

- a is x_{n-i} with $i = p + k$ and $k \geq 1$. Both sides are convertible to `k-1+p`.

□

Corollary 4.5 *Let a be a term or a proposition of Z_b, b be a term and l be a list of variables.*

$$F(\{b/x\}a, l) \leftrightarrow^*_{\sigma\in} F(a, x \cdot l)[F(b, l) \cdot \mathtt{id}]$$

Definition 4.6 *Let P be a proposition of Z_b. $F_\varepsilon(P)$ denotes the proposition $\varepsilon(F(P))$ of $\in_{es}$.*

Proposition 4.7 *For any propositions A and B of Z_b,*

$$\begin{array}{llll} F_\varepsilon(A \Rightarrow B) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & F_\varepsilon(A) \Rightarrow F_\varepsilon(B) \\ F_\varepsilon(A \vee B) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & F_\varepsilon(A) \vee F_\varepsilon(B) \\ F_\varepsilon(A \wedge B) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & F_\varepsilon(A) \wedge F_\varepsilon(B) \\ F_\varepsilon(\neg A) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & \neg F_\varepsilon(A) \\ F_\varepsilon(\bot) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & \bot \\ F_\varepsilon(\forall x\, A) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & \forall x\, F_\varepsilon(A) \\ F_\varepsilon(\exists x\, A) & \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} & \exists x\, F_\varepsilon(A) \end{array}$$

Proof. We only consider the case $\mathrm{F}_\varepsilon(\forall x\, A) \leftrightarrow^*_{\sigma\in} \to_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} \forall x\, \mathrm{F}_\varepsilon(A)$.

$$\begin{array}{rll} & \varepsilon(F(\forall x\, P)) & \\ = & \varepsilon(\dot\forall F(P, x)) & \\ \to_{\mathcal{L}} & \forall y\, \varepsilon(F(P, x)[\![y \cdot \mathtt{id}]\!]) & \\ = & \forall y\, \varepsilon(F(P, x)[\![F(y) \cdot \mathtt{id}]\!]) & \\ \leftrightarrow^*_{\sigma\in} & \forall y\, \varepsilon(F(\{y/x\}P)) & \text{(corollary 4.5)} \\ \leftrightarrow^*_{\sigma\in} & \forall y\, \varepsilon(\{y/x\}F(P)) & \text{(lemma 4.3)} \\ = & \forall x\, \varepsilon(F(P)) & \end{array}$$

□

4.2 Preservation of provability

Proposition 4.8 *If a sequent of* Z_b *is provable (in sequent calculus) then the* F_ε*-translation of this sequent is provable (in* $\in_{es}$*).*

Proof. Let π be a proof of $\Gamma \vdash \Delta$, we show by induction on π that there exists a proof of $\mathrm{F}_\varepsilon(\Gamma) \vdash \mathrm{F}_\varepsilon(\Delta)$ in $\in_{es}$.

In the case of an application of the rule $\forall$-l, π has the form:

$$\dfrac{\dfrac{\pi_0}{\Gamma, \{t/x\}P \vdash \Delta}}{\Gamma, \forall x\, P \vdash \Delta}\ \forall\text{-l}$$

By induction hypothesis there exists a proof π' such that

$$\dfrac{\pi'}{\mathrm{F}_\varepsilon(\Gamma), \mathrm{F}_\varepsilon(\{t/x\}P) \vdash \mathrm{F}_\varepsilon(\Delta)}$$

Since, according to lemma 4.3,

$$\varepsilon(F(\{t/x\}P)) \leftrightarrow^*_{\sigma_\in}\ \varepsilon(\{F(t)/x\}F(P))$$

we can extend π' with the rule:

$$\dfrac{\mathrm{F}_\varepsilon(\Gamma), \{F(t)/x\}\mathrm{F}_\varepsilon(P) \vdash \mathrm{F}_\varepsilon(\Delta)}{\mathrm{F}_\varepsilon(\Gamma), \forall x\, \mathrm{F}_\varepsilon(P) \vdash \mathrm{F}_\varepsilon(\Delta)}\ \forall\text{-l}\ (x, \mathrm{F}_\varepsilon(P), \mathrm{F}(t))$$

According to proposition 4.7: $\forall x\, \mathrm{F}_\varepsilon(P) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}}\ \mathrm{F}_\varepsilon(\forall x\, P)$; so we have obtained a proof of the sequent: $\mathrm{F}_\varepsilon(\Gamma), \mathrm{F}_\varepsilon(\forall x\, P) \vdash \mathrm{F}_\varepsilon(\Delta)$

In the case of the rule $\forall$-r the argument is the same except that a variable y is used instead of t. The cases for the other rules are similar or trivial. □

As a consequence the following lemma holds:

Lemma 4.9 *If* P *is a proposition of* Z_b *provable under the axioms* $\mathcal{A}$*, then there exists a proof of* $F_\varepsilon(P)$*, in* $\in_{es}$*, under the axioms* $F_\varepsilon(\mathcal{A})$*.*

4.3 Proof of conservativity

The converse of lemma 4.9 is more difficult to establish since a proof in $\in_{es}$ may contain terms or propositions not being F-translations.

Example 4.10 *We give in* $\in_{es}$ *a proof* π *of the F-translation of the sequent* $\forall x\ x = x \vdash \exists y\ y = y$*.*

$$\dfrac{\dfrac{\varepsilon(z[\uparrow] \doteq z[\uparrow]) \vdash \varepsilon(z[\uparrow] \doteq z[\uparrow])}{\forall x\ \varepsilon(x \doteq x) \vdash \varepsilon(z[\uparrow] \doteq z[\uparrow])}\ \forall\text{-}l\ (x, \varepsilon(x \doteq x), z[\uparrow])}{\forall x\ \varepsilon(x \doteq x) \vdash \exists y\ \varepsilon(y \doteq y)}\ \exists\text{-r}\ (y, \varepsilon(y \doteq y), z[\uparrow])$$

Here the proposition $\varepsilon(z[\uparrow] \doteq z[\uparrow])$ *is not a* F_ε*-translation, but we note that we can replace all occurrences of the term* $z[\uparrow]$ *by the term* z *and then all propositions and witnesses of this proof are in the image of F. It is then easy to deduce, in* Z_b*, a proof, isomorphic to* π*, of the sequent* $\forall x\ x = x \vdash \exists y\ y = y$*.*

We are going to prove that given a cut-free proof π in $\in_{es}$ of the translation of a sequent $\Gamma \vdash \Delta$ from Z_b, it is possible to transform π into a proof where all propositions and witnesses are in the image of F; since this transformation does not change the structure of the proof, it is then simple to translate the proof into an isomorphic proof of $\Gamma \vdash \Delta$ in Z_b.

The reason why this transformation applied to a proof still gives a proof is that on the one hand terms are not in the image of the translation only because some occurrences of their variables are not in the image of the translation, and on the other hand these same occurrences are only appearing as witnesses in the quantifier rules; in other words such occurrences can't be bound. This way the

transformation consists in replacing these occurrences altogether with the environment they appear under (as in the example).

We now define the transformation function ϕ on the propositions/terms in normal form for the rewrite relation $\rightarrow_{\sigma_\in}$. We first need to characterize terms in normal form.

Lemma 4.11 *Partial characterization of the terms of sort o or* set *in normal form for $\rightarrow_{\sigma_\in}$.*

A term in normal form has one of the following forms:

- $\mathtt{i}$ *where* $\mathtt{i}$ *is an index.*
- $\mathtt{i}[Y]$ *where* $\mathtt{i}$ *is an index and Y a variable of sort* subst*.*
- $\mathtt{i}[Y \circ s]$ *where* $\mathtt{i}$ *is an index, Y a variable of sort* subst *and s a term of sort* subst *in normal form distinct from* $\mathtt{id}$*.*
- x *where x is a variable of sort* set *or o.*
- $x[s]$ *where x is a variable of sort* set *and s a term of sort* subst*.*
- $x[\![s]\!]$ *where x is a variable of sort o and s a term of sort* subst*.*
- c *where c is a constant of sort* set *or o.*
- $f(t_1, \ldots, t_n)$ *where f is a function symbol of rank $(s_1, \ldots, s_n)s$ with s_i and $s \in \{set, o\}$ and where each t_i is term of sort s_i in normal form.*

Proof. Simple induction on the definition of terms. □

4.3.1 The function ϕ and its properties

ϕ has been designed to be a projection onto the image of F.

Definition 4.12 *We define a function ϕ on normal forms, for $\rightarrow_{\sigma_\in}$, of terms of sorts* set *and o and then we extend this definition to propositions.*

Let X_0 be a variable of sort set*. We are now assuming that X_0 is never used as a bound variable.* $\mathtt{i}$ *denotes an index, c denotes a constant, f denotes a function symbol whose binding arity is $(a_1, \ldots, a_p)$, and P,Q denote propositions.*

$$\begin{array}{lcl}
\phi_n(x_{set}) = \phi_n(x_{set}[s]) & = & \left\{ \begin{array}{ll} x_{set} & \text{if } n = 0 \\ x_{set}[\uparrow^n] & \text{if } n > 0 \end{array} \right. \\
\phi_n(\mathtt{i}) & = & \left\{ \begin{array}{ll} X_0 & \text{if } n = 0 \\ \mathtt{i} & \text{if } n > 0 \text{ and } i \leq n \\ X_0[\uparrow^n] & \text{if } i > n \geq 1 \end{array} \right. \\
\phi_n(x_o) = \phi_n(x_o[\![s]\!]) & = & \dot{\perp}
\end{array}$$

$$\begin{array}{lcl}
\phi_n(c) & = & c \\
\phi_n(f(t_1, \ldots, t_p)) & = & f(\phi_{n+a_1}(t_1), \ldots, \phi_{n+a_p}(t_p))
\end{array}$$

$$\begin{array}{lcl}
\phi(\varepsilon(t)) & = & \varepsilon(\phi_0(t)) \\
\phi(P \Rightarrow Q) & = & \phi(P) \Rightarrow \phi(Q) \\
\phi(P \wedge Q) & = & \phi(P) \wedge \phi(Q) \\
\phi(P \vee Q) & = & \phi(P) \vee \phi(Q) \\
\phi(\neg P) & = & \neg\phi(P) \\
\phi(\forall x\, P) & = & \forall x\, \phi(P) \\
\phi(\exists x\, P) & = & \exists x\, \phi(P)
\end{array}$$

$\phi(t)$ denotes $\phi_0(t)$ when t is a term. Since any term/proposition has a unique normal form, we consider that ϕ is defined on any term/proposition.

The following proposition shows that ϕ is invariant on the image of F.

Proposition 4.13 *If P is a proposition of Z_b then*
$$\phi(F(P)) = F(P) \text{ and } \phi(F_\varepsilon(P)) = F_\varepsilon(P).$$

Proof. Simple induction on P. □

The following lemma shows that the image of ϕ for atomic propositions is in the image of F.

Proposition 4.14 *Let t be a term of sort* set *(resp. of sort o) in normal form, n be a natural number, and $x_1 \dots x_n$ be variables not occurring in t and distinct from X_0. There exists a unique term (resp. proposition) u of Z_b such that*
$$F(u, x_n \dots x_1) = \phi_n(t)$$

Proof. Simple induction on t. □

Lemma 4.15 *Let t be a term of sort o or* set*, n be a positive natural number and p be natural number.*
$$\phi_{n+p}(t[\Uparrow^p(\uparrow^n)]) \leftrightarrow^*_{\sigma\in} \phi_p(t)[\Uparrow^p(\uparrow^n)]$$

Proof. By induction on t, for all p.

- t has the form $f(u_1, \dots, u_k)$ where f is a symbol of binding arity $a_1, \dots, a_k$.

  We have
  $$\phi_{n+p}(f(\dots, u_i, \dots)[\Uparrow^p(\uparrow^n)]) \to_{\sigma\in} f(\dots, \phi_{n+p+a_i}(u_i[\Uparrow^{p+a_i}(\uparrow^n)]), \dots)$$
  and
  $$\phi_p(f(\dots, u_i, \dots))[\Uparrow^p(\uparrow^n)] \to_{\sigma\in} f(\dots, \phi_{p+a_i}(u_i)[\Uparrow^{p+a_i}(\uparrow^n)], \dots)$$
  We conclude using the induction hypothesis: for each i,
  $$\phi_{n+p+a_i}(u_i[\Uparrow^{p+a_i}(\uparrow^n)]) \leftrightarrow^*_{\sigma\in} \phi_{p+a_i}(u_i)[\Uparrow^{p+a_i}(\uparrow^n)]$$

- t has the form $x[s]$.

  We have
  $$\phi_{n+p}(x[s][\Uparrow^p(\uparrow^n)]) \to_{\sigma\in} \phi_{n+p}(x[s \circ \Uparrow^p(\uparrow^n)]) = x[\uparrow^{n+p}]$$
  and
  $$\phi_p(x[s])[\Uparrow^p(\uparrow^n)] = x[\uparrow^p][\Uparrow^p(\uparrow^n)] \to^*_{\sigma\in} x[\uparrow^{n+p}].$$

- t is an index `i`.

  When $p < i$ we have
  $$\phi_{n+p}(\mathtt{i}[\Uparrow^p(\uparrow^n)]) \to^*_{\sigma\in} \phi_{n+p}(\mathtt{i+n+p}) = X_0[\uparrow^{n+p}]$$
  and
  $$\phi_p(\mathtt{i})[\Uparrow^p(\uparrow^n)] = X_0[\uparrow^p][\Uparrow^p(\uparrow^n)] \to^*_{\sigma\in} X_0[\uparrow^{n+p}]$$
  Otherwise when $i \leq p$ we have
  $$\phi_{n+p}(\mathtt{i}[\Uparrow^p(\uparrow^n)]) \to^*_{\sigma\in} \phi_{n+p}(\mathtt{i}) = \mathtt{i}$$
  and
  $$\phi_p(\mathtt{i})[\Uparrow^p(\uparrow^n)] = \mathtt{i}[\Uparrow^p(\uparrow^n)] \to^*_{\sigma\in} \mathtt{i}$$

□

Corollary 4.16 *Let t be a term of sort o or* set *and n be a positive natural number.*
$$\phi_n(t[\uparrow^n]) \leftrightarrow^*_{\sigma\in} \phi(t)[\uparrow^n]$$

Lemma 4.17 *Let t be a term of sort o or* set*, a be a term of sort* set *and n be a natural number.*

$$\phi_n(t[\Uparrow^n(a \cdot \mathtt{id})]) \leftrightarrow^*_{\sigma\in} \phi_{n+1}(t)[\Uparrow^n(\phi(a) \cdot \mathtt{id})]$$

Proof. The proof follows the argument of the previous one. The only particular case is when t is the index `n+1`: the following two series of equations hold

$$\phi_n(\mathtt{n+1}[\Uparrow^n(a \cdot \mathtt{id})]) \rightarrow^*_{\sigma\in} \phi_n(a[\uparrow^n])$$

$$\phi_{n+1}(\mathtt{n+1})[\Uparrow^n(\phi(a) \cdot \mathtt{id})] = \mathtt{n+1}[\Uparrow^n(\phi(a) \cdot \mathtt{id})] \rightarrow^*_{\sigma\in} \phi(a)[\uparrow^n]$$

and we conclude using corollary 4.16:

$$\phi_n(a[\uparrow^n]) \leftrightarrow^*_{\sigma\in} \phi(a)[\uparrow^n]$$

□

Corollary 4.18 *Let t be a term of sort o or* set *and a be a term of sort* set*.*

$$\phi(t[a \cdot \mathtt{id}]) \leftrightarrow^*_{\sigma\in} \phi_1(t)[\phi(a) \cdot \mathtt{id}]$$

Proposition 4.19 *Let $\{t_i/x_i\}$ be a substitution where each x_i is a variable of sort* set *and let t be a term or a proposition of Z_b.*

$$\phi(\{t_i/x_i\}F(t)) = \{\phi(t_i)/x_i\}F(t)$$

Proof. This proposition is a corollary of the following more general proposition: *Let $\{t_i/x_i\}$ be a substitution where each x_i is a variable of sort* set *and let t be a term or a proposition of Z_b; Let n be a natural number and l a list of variables whose length is n. $\phi_n(\{t_i/x_i\}F(t,l)) = \{\phi_0(t_i)/x_i\}F(t,l)$.* The proof is done by induction on t and uses corollary 4.16 when t is $x[\uparrow^n]$. □

Lemma 4.20 (ϕ is compatible with $\mathcal{L}$)
Let a and b be propositions. If $a \rightarrow_{\mathcal{L}} b$ then $\phi(a) \leftrightarrow^*_{\sigma\in} \rightarrow_{\mathcal{L}} \leftrightarrow^*_{\sigma\in} \phi(b)$.

Proof. We are reasoning by case analysis on the head rewrite step.
When a is $\varepsilon(\dot\forall t)$ and b is $\forall x\ \varepsilon(t[\![x \cdot \mathtt{id}]\!])$ we have

$$\begin{array}{rcll}
\phi(\varepsilon(\dot\forall t)) & = & \varepsilon(\phi(\dot\forall t)) & \\
& = & \varepsilon(\dot\forall \phi_1(t)) & \\
& \rightarrow_{\mathcal{L}} & \forall x\ \varepsilon(\phi_1(t)[\![x \cdot \mathtt{id}]\!]) & \\
& = & \forall x\ \varepsilon(\phi_1(t)[\![\phi(x) \cdot \mathtt{id}]\!]) & \\
& \leftrightarrow^*_{\sigma\in} & \forall x\ \varepsilon(\phi(t[\![x \cdot \mathtt{id}]\!])) & \text{(corollary 4.18)} \\
& = & \phi(\forall x\ \varepsilon(t[\![x \cdot \mathtt{id}]\!])) &
\end{array}$$

The other cases are similar. □

4.3.2 Transformation of a proof in $\in_{es}$

Definition 4.21

1. *A F-term is a term (of sort* set *or o) in the image of F.*
2. *A F-proposition is a proposition of the form $\varepsilon(t)$ where t is a F-term of sort o.*
3. *A F^τ_ε-proposition is a proposition of the form $\sigma F_\varepsilon(P)$ where σ is a substitution the domain of which contains only variables of sort* set*.*

Lemma 4.22 *Let $\Gamma \vdash \Delta$ be a sequent where each proposition is a F^τ_ε-proposition and which has a cut-free proof π. There exists a proof of the sequent $\phi(\Gamma) \vdash \phi(\Delta)$ where all propositions are F-propositions and all witnesses are F-terms.*

Proof. By induction on π. We only give the details for the two important cases.

- In the case of an application of the rule $\forall$-l, π has the form:

$$\frac{\dfrac{\pi_0}{\Gamma, \{t/x\}P' \vdash \Delta}}{\Gamma, Q' \vdash \Delta}\,\forall\text{-l}\ (x, P', t) \qquad \text{with } Q' \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \forall x\ P'$$

We know that Q' is a $\mathrm{F}^\tau_\varepsilon$-proposition so there exists a substitution τ the variables of its domain being of sort *set* and a proposition Q_Z of Z_b such that $Q' \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \tau\mathrm{F}_\varepsilon(Q_Z)$; so $\forall x\, P' \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \tau\mathrm{F}_\varepsilon(Q_Z)$; there can be only one possibility: Q_Z has the form $\forall x\, Q'_Z$. We have now $\forall x\, P' \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \tau\mathrm{F}_\varepsilon(\forall x\, Q'_Z) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \tau\forall x\mathrm{F}_\varepsilon(Q'_Z) = \forall x\, \tau\mathrm{F}_\varepsilon(Q'_Z)$ and so $P' \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \tau\mathrm{F}_\varepsilon(Q'_Z)$.

We have just proved that π_0 is a proof of the sequent:

$$\Gamma, \{t/x\}\tau\mathrm{F}_\varepsilon(Q'_Z) \vdash \Delta \qquad (S_1)$$

where all propositions are F-propositions and all witnesses are F-terms.

Since x was bound in $\tau\forall x\,\mathrm{F}_\varepsilon(Q'_Z)$ we know that $x \notin Var(\tau)$ so
$$\{t/x\}\tau\mathrm{F}_\varepsilon(Q'_Z) = \{t/x;\tau\}\mathrm{F}_\varepsilon(Q'_Z).$$

(S_1) is then a $\mathrm{F}^\tau_\varepsilon$-sequent and we can apply the induction hypothesis: there exists a proof π' of the sequent
$$\phi(\Gamma), \phi(\{t/x;\tau\}\mathrm{F}_\varepsilon(Q'_Z)) \vdash \phi(\Delta) \qquad (S'_1)$$

According to proposition 4.19,

$$\phi(\{t/x;\tau\}\mathrm{F}_\varepsilon(Q'_Z)) = \{\phi(t)/x;\phi(\tau)\}\mathrm{F}_\varepsilon(Q'_Z)$$

x not being in $Ran(\tau)$ we deduce

$$\{\phi(t)/x\,;\,\phi(\tau)\}\mathrm{F}_\varepsilon(Q'_Z) = \{\phi(t)/x\}\phi(\tau)\mathrm{F}_\varepsilon(Q'_Z)$$

π' can then be extended in the following way:

$$\frac{\dfrac{\pi'}{\phi(\Gamma), \{\phi(t)/x\}\phi(\tau)\mathrm{F}_\varepsilon(Q'_Z) \vdash \phi(\Delta)}}{\phi(\Gamma), \forall x\, \phi(\tau)\mathrm{F}_\varepsilon(Q'_Z) \vdash \phi(\Delta)}\,\forall\text{-l}\ (x, \phi(\tau)\mathrm{F}_\varepsilon(Q'_Z), \phi(t))$$

$\forall x\, \phi(\tau)\mathrm{F}_\varepsilon(Q'_Z) = \phi(\tau)\forall x\,\mathrm{F}_\varepsilon(Q'_Z) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \phi(\tau)\mathrm{F}_\varepsilon(\forall x\ Q'_Z)$ holds, so we have just found a proof of the sequent $\phi(\Gamma), \phi(\tau)\mathrm{F}_\varepsilon(\forall x\, Q'_Z) \vdash \phi(\Delta)$ in which all propositions are F-propositions and all witnesses are F-terms.

- In the case of the rule axiom, π has the form:

$$\frac{}{\tau_1\mathrm{F}_\varepsilon(P) \vdash \tau_2\mathrm{F}_\varepsilon(Q)}\,\text{axiom} \qquad \text{with } \tau_1\mathrm{F}_\varepsilon(P) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \tau_2\mathrm{F}_\varepsilon(Q)$$

According to lemma 4.20,
$$\phi(\tau_1\mathrm{F}_\varepsilon(P)) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \phi(\tau_2\mathrm{F}_\varepsilon(Q))$$

and according to proposition 4.19,

$$\phi(\tau_1\mathrm{F}_\varepsilon(P)) = \phi(\tau_1)\mathrm{F}_\varepsilon(P) \quad \text{and} \quad \phi(\tau_2\mathrm{F}_\varepsilon(Q)) = \phi(\tau_2)\mathrm{F}_\varepsilon(Q)$$

So we have

$$\frac{}{\phi(\tau_1)\mathrm{F}_\varepsilon(P) \vdash \phi(\tau_2)\mathrm{F}_\varepsilon(Q)}\,\text{axiom} \qquad \text{with } \phi(\tau_1)\mathrm{F}_\varepsilon(P) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \phi(\tau_2)\mathrm{F}_\varepsilon(Q)$$

□

Lemma 4.23 *For any sequent $\Gamma \vdash \Delta$ of Z_b, if there exists a proof π of its F-translation in $\in_{es}$ where all propositions are F-propositions and all witnesses are F-terms then there exists a proof of $\Gamma \vdash \Delta$ in Z_b.*

Proof. By induction on π. We only give the details for the rule $\forall$-l. Following the way the last proof was carried on, we know that π has the following form:

$$\dfrac{\dfrac{\pi_0}{\mathrm{F}_\varepsilon(\Gamma), \{F(t)/x\}\mathrm{F}_\varepsilon(P) \vdash \mathrm{F}_\varepsilon(\Delta)}}{\mathrm{F}_\varepsilon(\Gamma), \forall x\ \mathrm{F}_\varepsilon(P) \vdash \mathrm{F}_\varepsilon(\Delta)}\ \forall\text{-l}\ (x, \mathrm{F}_\varepsilon(P), F(t))$$

According to lemma 4.3, $\{F(t)/x\}\mathrm{F}_\varepsilon(P) \leftrightarrow^*_{\sigma \in \cup \mathcal{L}} \mathrm{F}_\varepsilon(\{t/x\}P)$; so we apply the induction hypothesis and we have a proof π' of the sequent:

$$\Gamma, \{t/x\}P \vdash \Delta$$

Since, according to proposition 4.7, $\forall x\ \mathrm{F}_\varepsilon(P) \leftrightarrow^*_{\sigma \in \cup \mathcal{L}} \mathrm{F}_\varepsilon(\forall x\ P)$, the proof

$$\dfrac{\dfrac{\pi'}{\Gamma, \{t/x\}P \vdash \Delta}}{\Gamma, \forall x\ P \vdash \Delta}\ \forall\text{-l}$$

is the one we were looking for. □

Theorem 4.24 *Let $\mathcal{A}$ be a theory expressed in the language Z_b. The F_ε-translation of $\mathcal{A}$ into the system $\in_{es}$ is equivalent to $\mathcal{A}$.*

Proof.

1. Let P be a proposition of Z_b provable in sequent calculus under the axioms $\mathcal{A}$ then, according to lemma 4.9, $\mathrm{F}_\varepsilon(P)$ is provable in $\in_{es}$ under the axioms $\mathrm{F}_\varepsilon(\mathcal{A})$.
2. Let P be a proposition of Z_b such that $\mathrm{F}_\varepsilon(P)$ is provable in $\in_{es}$ under the axioms $\mathrm{F}_\varepsilon(\mathcal{A})$. By definition there exists a formula list Γ of $\mathcal{A}$ such that $\mathrm{F}_\varepsilon(\Gamma) \vdash \mathrm{F}_\varepsilon(P)$ is provable. According to theorem 3.15, there exists a cut-free proof of $\mathrm{F}_\varepsilon(\Gamma) \vdash \mathrm{F}_\varepsilon(P)$. According to lemma 4.22, there exists a proof π of $\mathrm{F}_\varepsilon(\Gamma) \vdash \mathrm{F}_\varepsilon(P)$ where all propositions are F-propositions and all witnesses are F-terms. According to lemma 4.23, there exists a proof of $\Gamma \vdash P$. We can now conclude that there exists a proof of P under the axioms $\mathcal{A}$.

□

Now, we have to prove that the translation of the axiom schemes of Z_b is equivalent to a (finite) sub-theory of Z_{es}.

4.4 Coding axiom schemes into $\in_{es}$

In this section we prove the equivalence, in $\in_{es}$, between the set composed of the translation of the two schemes of Z_b (equality and comprehension schemes) and a set composed of the two axioms A_1 and A_2 of Z_{es} given below.

The two schemes of Z_b are:

($\mathcal{S}_1$) the comprehension scheme:

For any proposition P whose free variables are among $x_1, \ldots, x_n, z$,
$\forall x_1\ \ldots \forall x_n\ \forall y \forall z (z \in \{z \in y \mid P\} \Leftrightarrow (z \in y \wedge P))$

($\mathcal{S}_2$) the equality scheme:

For any proposition P whose free variables are among $x_1, \ldots, x_n, z$,
$\forall x_1\ \ldots \forall x_n\ \forall x \forall y (x = y \Rightarrow (\{x/z\}P \Rightarrow \{y/z\}P))$

The two axioms of Z_{es} we consider are:

(A_1) $\quad \forall p \forall y \forall z\ (\varepsilon(y \dot{\in} \{z \mid p\}) \Leftrightarrow (\varepsilon(y \dot{\in} z) \wedge \varepsilon(p[\![y \cdot \mathtt{id}]\!])))$

(A_2) $\quad \forall p \forall x \forall y (\varepsilon(x \doteq y) \Rightarrow (\varepsilon(p[\![x \cdot \mathtt{id}]\!]) \Rightarrow \varepsilon(p[\![y \cdot \mathtt{id}]\!])))$

"Only if" direction We prove that a provable sequent (in $\in_{es}$) under the axioms $\mathrm{F}_\varepsilon(\mathcal{S}_1 \cup \mathcal{S}_2)$ is provable under the two axioms A_1 and A_2. In order to do that, it is sufficient to prove the following lemma which says that any proposition of $\mathrm{F}_\varepsilon(\mathcal{S}_i)$ is a consequence of A_i.

Lemma 4.25 *Let A be a proposition of $F_\varepsilon(\mathcal{S}_1)$ (resp. $F_\varepsilon(\mathcal{S}_2)$), the sequent $A_1 \vdash A$ (resp. $A_2 \vdash A$) is provable.*

Proof. Two cases are possible:

- If A is an axiom of $\mathrm{F}_\varepsilon(\mathcal{S}_1)$ then there exists a proposition P of Z_b whose free variables are among $x_1, \ldots, x_n, z$ and such that A has the form

  $$\mathrm{F}_\varepsilon(\forall x_1 \ldots \forall x_n \forall y \forall z\, (z \in \{z \in y \mid P\} \Leftrightarrow (z \in y \wedge P)))$$

  which, using proposition 4.7 and corollary 4.5, is convertible to

  $$\forall x_1 \ldots \forall x_n \forall y \forall z\, (\varepsilon(z \dot{\in} \{y \mid F(P,z)\}) \Leftrightarrow (\varepsilon(z \dot{\in} y) \wedge \varepsilon(F(P,z)[\![z \cdot \mathtt{id}]\!])))$$

  This proposition can be written

  $$\forall x_1 \ldots \forall x_n \{F(P,z)/Y\} \forall y \forall z\, (\varepsilon(z \dot{\in} \{y \mid Y\}) \Leftrightarrow (\varepsilon(z \dot{\in} y) \wedge \varepsilon(Y[\![z \cdot \mathtt{id}]\!])))$$

  since y and z do not occur in $F(P,z)$.

  Thus, to prove the sequent

  $$A_1 \vdash A$$

  we apply a sequence of n $\forall$-r rules and we then have to prove the sequent

  $$A_1 \vdash \{F(P,z)/Y\} \forall y \forall z\, \varepsilon(z \dot{\in} \{y \mid Y\}) \Leftrightarrow (\varepsilon(z \dot{\in} y) \wedge \varepsilon(Y[\![z \cdot \mathtt{id}]\!]))$$

  which is done using a $\forall_o$-l rule and the axiom rule.

- If A is an axiom of $\mathrm{F}_\varepsilon(\mathcal{S}_2)$ then there exists a proposition P of Z_b such that A has the form

  $$\mathrm{F}_\varepsilon(\forall x_1 \ldots \forall x_n \forall x \forall y (x = y \Rightarrow (\{x/z\}P \Rightarrow \{y/z\}P)))$$

  which is convertible to

  $$\forall x_1 \ldots \forall x_n \forall x \forall y (\varepsilon(x \doteq y) \Rightarrow (\varepsilon(F(P,z)[\![x \cdot \mathtt{id}]\!]) \Rightarrow \varepsilon(F(P,z)[\![y \cdot \mathtt{id}]\!])))$$

  the argument is then the same.

□

"If" direction Here we are following the same argument as in section 4.3.2: we are going to apply the function ϕ to a proof that contains instances of axioms A_1 and A_2. Previously this has been possible since in a cut-free proof all propositions had the form $\sigma\mathrm{F}_\varepsilon(P)$ and so $\phi(\sigma\mathrm{F}_\varepsilon(P))$ was convertible to $\phi(\sigma)\phi(\mathrm{F}_\varepsilon(P))$. Here this invariant is not true but a finer one holds: each occurrence (in the proposition) of a variable in the domain of σ occurs under an operator $[\uparrow^n]$ for a suitable n (such an occurrence is said to be pre-cooked).

First we give the formal definition of the well-formedness criterion and prove that whenever a sequent S for which that criterion holds, $\phi(S)$ has a proof which is in the image of the F-translation. Then we apply this result to proofs involving the encodings of the schemes.

The following predicate Pwf_V expresses a criterion of well-formedness: it holds for any proposition P such that all bound occurrences of variables of sort *set* and all free occurrences of variables of V (of sort *set*) are pre-cooked. This predicate is defined on normal forms for the relation $\rightarrow_{\sigma\in}$. Twf^n_V is its counterpart on terms.

Definition 4.26 *We define* $\mathrm{Twf}_V^n(t)$ *where* t *is a term of sort* o *or* set *in normal form for* $\rightarrow_{\sigma_\in}$, V *a set of variables of sort* set *and* n *a natural number.*

f *denotes a function symbol whose binding arity is* $(a_1, \ldots, a_k)$.

$\mathrm{Twf}_V^n(x_{set}[s])$ *holds if* $x \in V$ *implies that* s *has the form* $\uparrow^n$.
$\mathrm{Twf}_V^n(x_{set})$ *holds if* $x \in V$ *implies* $n = 0$.
$\mathrm{Twf}_V^n(x_o)$ *holds.*
$\mathrm{Twf}_V^n(x_o[\![s]\!])$ *holds.*
$\mathrm{Twf}_V^n(\mathtt{i})$ *holds (where* $\mathtt{i}$ *is an index).*
$\mathrm{Twf}_V^n(f(t_1, \ldots, t_k))$ *holds if for each* t_i, $\mathrm{Twf}_V^{n+a_i}(t_i)$ *holds.*
$\mathrm{Twf}_V^n(c)$ *holds (where* c *is constant of sort* set *or* o*).*

Definition 4.27 *We define* $\mathrm{Pwf}_V(P)$ *where* P *is a proposition and* V *a set of variables of sort* set.

$\mathrm{Pwf}_V(\varepsilon(p))$ *holds if* $\mathrm{Twf}_V^0(p)$ *holds.*
$\mathrm{Pwf}_V(A \Rightarrow B)$ *holds if* $\mathrm{Pwf}_V(A)$ *and* $\mathrm{Pwf}_V(B)$ *hold.*
$\mathrm{Pwf}_V(A \wedge B)$ *holds if* $\mathrm{Pwf}_V(A)$ *and* $\mathrm{Pwf}_V(B)$ *hold.*
$\mathrm{Pwf}_V(A \vee B)$ *holds if* $\mathrm{Pwf}_V(A)$ *and* $\mathrm{Pwf}_V(B)$ *hold.*
$\mathrm{Pwf}_V(\neg A)$ *holds if* $\mathrm{Pwf}_V(A)$ *holds.*
$\mathrm{Pwf}_V(\bot)$ *holds.*
$\mathrm{Pwf}_V(\forall_{set} x\ P)$ *holds if* $\mathrm{Pwf}_{V\cup\{x\}}(P)$ *holds.*
$\mathrm{Pwf}_V(\exists_{set} x\ P)$ *holds if* $\mathrm{Pwf}_{V\cup\{x\}}(P)$ *holds.*

For any proposition P, $\mathrm{Pwf}(P)$ *denotes* $\mathrm{Pwf}_\emptyset(P)$.

For example $\forall y\, \varepsilon(\dot\forall(x[\uparrow] \doteq y[\uparrow]))$ is a F-proposition whereas $\forall y\, \varepsilon(\dot\forall(x \doteq y[\uparrow]))$ is not. Pwf holds for both propositions and Pwf_x holds only for the first one.

Lemma 4.28 *Let* x *be a variable of sort* set, V *be a set of variables of sort* set, P *be a proposition and* t *be a term of sort* set. *If* $\mathrm{Pwf}_{V\cup\{x\}}(P)$ *holds then* $\mathrm{Pwf}_V(P)$ *holds and if* $\mathrm{Pwf}(P)$ *holds then* $\mathrm{Pwf}(\{t/x\}P)$ *holds.*

Proof. Trivial. □

Lemma 4.29 *Let* t *and* u *be terms of sort* set *and* x *be a variable of sort* set. *For any natural number* n, *if* $\mathrm{Twf}_x^n(u)$ *holds then* $\phi_n(\{t/x\}u) \leftrightarrow^*_{\sigma_\in} \{\phi(t)/x\}\phi_n(u)$.

Proof. By induction on u, for all n. The proof is almost the same as in proposition 4.19 except when u is $x[\uparrow^p]$ with $p \neq n$, $\mathrm{Twf}_x^n(x[\uparrow^p])$ is false.

□

Lemma 4.30 *If* $\mathrm{Pwf}_x(P)$ *then* $\phi(\{t/x\}P) \leftrightarrow^*_{\sigma_\in} \{\phi(t)/x\}\phi(P)$.

Proof. Simple induction on P, lemma 4.29 is used in the case of an atomic proposition. □

Lemma 4.31 *Let* $\Gamma \vdash \Delta$ *be a sequent of* $\in_{es}$ *for which* Pwf *holds. Let* π *be a cut-free proof of this sequent. There exists a proof of* $\phi(\Gamma \vdash \Delta)$, *where all propositions are F-propositions and all witnesses are F-terms.*

Proof. By induction on π. We only give the details for the two important cases.

- In the case of an application of the rule $\forall$-l, π has the form:

$$\dfrac{\dfrac{\pi_0}{\Gamma, \{t/x\}P \vdash \Delta}}{\Gamma, Q \vdash \Delta}\ \forall\text{-l}\ (x, P, t) \qquad \text{with } Q \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \forall x\, P$$

  By hypothesis, $\mathrm{Pwf}(Q)$ holds (*i.e.* $\mathrm{Pwf}(\forall x\, P)$ holds) so $\mathrm{Pwf}_x(P)$ holds and then $\mathrm{Pwf}(\{t/x\}P)$ holds. We can then apply the induction hypothesis: there exists a proof π' of the sequent

$$\phi(\Gamma), \phi(\{t/x\}P) \vdash \phi(\Delta)$$

where all propositions are F-propositions and all witnesses are F-terms.

Since $\mathrm{Pwf}_x(P)$ holds, according to lemma 4.30 we have $\phi(\{t/x\}P) \leftrightarrow^*_{\sigma_\in} \{\phi(t)/x\}\phi(P)$ and we can write:

$$\dfrac{\dfrac{\pi'}{\phi(\Gamma), \{\phi(t)/x\}\phi(P) \vdash \phi(\Delta)}}{\phi(\Gamma), \forall x\, \phi(P) \vdash \phi(\Delta)}\ \forall\text{-l}\ (x, \phi(P), \phi(t))$$

According to proposition 4.14 there exists a term t_Z of Z_b such that $F(t_Z) = \phi(t)$ and according to propositions 4.14 and 4.7 there exists a proposition P_Z of Z_b such that $\mathrm{F}_\varepsilon(P_Z) = \phi(P)$. We conclude using proposition 4.7: $\forall x\, \phi(P) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \mathrm{F}_\varepsilon(\forall x\, P_Z)$.

- In the case of the rule axiom, π has the form:

$$\dfrac{}{P \vdash Q}\ \text{axiom} \qquad \text{with } P \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} Q$$

According to lemma 4.20, $\phi(P) \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \phi(Q)$. We conclude using the fact that $\phi(P)$ and $\phi(Q)$ are convertible to F-propositions (proposition 4.14).

□

We are now dealing with the schemes of Z_b.

Definition 4.32 *The axioms A_1 and A_2 have the form $\forall Y A'_1$ and $\forall Y A'_2$, so:*

1. *An instance of A_1 (resp. A_2) is any proposition that has the form $\{t/Y\}A'_1$ (resp. $\{t/Y\}A'_2$).*
2. *$\mathcal{B}$ is the set of propositions P such that: $P \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \bar{\forall}\phi(Q)$ where Q is an instance of A_1 or A_2.*

Lemma 4.33

1. Let t be a term of sort set *or o, n be a natural number and $x_1, \ldots, x_n$ be variables (of sort* set*) without occurrence in t. For any natural number k, $\mathrm{Twf}^k_{x_1,\ldots,x_n}$ holds for $t[\Uparrow^k(x_n \cdot \ldots \cdot x_1 \cdot \mathtt{id})]$.*

2. Let t be a term of sort set *or o, n and k be natural numbers, $x_1, \ldots, x_n$ be variables and y be a variable without occurrence in t. If $\mathrm{Twf}^k_{x_1,\ldots,x_n}$ holds for $t[y \cdot \mathtt{id}]$ then $\mathrm{Twf}^{k+1}_{x_1,\ldots,x_n}$ holds for t.*

Proof. Simple induction on t, for all k. □

Proposition 4.34 *Let P and P' be propositions such that $P \to_{\mathcal{L}} P'$. Then for any set of variables V, Pwf_V holds for P if and only if it holds for P'.*

Proof. Consequence of the preceding lemma. □

Lemma 4.35 Pwf *holds for any instance of A_1 and for any instance of A_2.*

Proof. We only process the case of the comprehension scheme; the argument being the same for the equality scheme.

Let t be a term of sort o and z a variable of sort *set* which does not occur free in t. According to the preceding lemma, we know that Twf^0_z holds for $t[\![z \cdot \mathtt{id}]\!]$.

Let y be a variable that does not occur free in t; hence $\mathrm{Twf}^0_{y,z}$ holds for $t[\![z \cdot \mathtt{id}]\!]$. Now, we have that $\mathrm{Pwf}_{y,z}$ holds for $\varepsilon(t[\![z \cdot \mathtt{id}]\!])$.

Then we conclude that Pwf holds for $\forall y \forall z\ (\varepsilon(z \mathrel{\dot\in} \{y \mid t\}) \Leftrightarrow (\varepsilon(z \mathrel{\dot\in} y) \wedge \varepsilon(t[\![z \cdot \mathtt{id}]\!])))$.

□

Lemma 4.36 *Let L be a list of propositions of $\{A_1, A_2\}$ and Γ, Δ be lists of propositions for which* Pwf *holds. If $L, \Gamma \vdash \Delta$ has a cut-free proof π then there exists a list M of propositions of $\mathcal{B}$ and a proof of the sequent $M, \phi(\Gamma) \vdash \phi(\Delta)$.*

Proof. By induction on π.

- Rule for which the principal formula is in Δ or Γ.

  The argument is the same as in the proof of the lemma 4.31.

- Rule for which the principal formula is in L: it is necessarily a rule $\forall$-l. So π has the form:

$$\dfrac{\dfrac{\pi_0}{\{t/Y\}P, L', \Gamma \vdash \Delta}}{\forall_o Y\ P, L', \Gamma \vdash \Delta}\ \forall_o\text{-l}\ (Y, P, t)$$

  $\forall_o Y\ P$ being one of the axioms $A_1\ A_2$, according to lemma 4.35, Pwf holds for $\{t/Y\}P$. We can apply the induction hypothesis: there exists a list M of propositions of $\mathcal{B}$ and a proof π' of the sequent $M, \phi(\{t/Y\}P), \phi(\Gamma) \vdash \phi(\Delta)$. We then extend π' with a sequence of applications of rules $\forall$-l in such a way that the result is a proof of $M, \bar{\forall}\phi(\{t/Y\}P), \phi(\Gamma) \vdash \phi(\Delta)$.

  We then check that $\bar{\forall}\phi(\{t/Y\}P)$ is a proposition of $\mathcal{B}$.

- In the case of an application of a weakening rule or of a left contraction rule on a formula of L we just apply the induction hypothesis.

□

Lemma 4.37 $\mathcal{B} \subseteq F_\varepsilon(\mathcal{S}_1 \cup \mathcal{S}_2)$

Proof. Let P be proposition of $\mathcal{B}$. If P has been obtained from A_1, there exists a term t of sort o such that

$$P \leftrightarrow^*_{\sigma_\in \cup \mathcal{L}} \bar{\forall}\,\phi(\forall y \forall z\ (\varepsilon(z \dot{\in} \{y \mid t\}) \Leftrightarrow (\varepsilon(z \dot{\in} y) \wedge \varepsilon(t[\![z \cdot \mathtt{id}]\!]))))$$

and we note that since propositions are considered up to α-conversion, y and z are not occurring in t.

Applying the definition of ϕ, this proposition is equal to

$$\bar{\forall}\forall y \forall z\ (\varepsilon(z \dot{\in} \{y \mid \phi_1(t)\}) \Leftrightarrow (\varepsilon(z \dot{\in} y) \wedge \varepsilon(\phi(t[\![z \cdot \mathtt{id}]\!]))))$$

According to corollary 4.18, $\phi(t[\![z \cdot \mathtt{id}]\!]) \leftrightarrow^*_{\sigma_\in} \phi_1(t)[\![z \cdot \mathtt{id}]\!]$; since $z \notin FV(t)$, according to proposition 4.14, there exists a proposition P_Z of Z_b such that, $\phi_1(t) \leftrightarrow^*_{\sigma_\in} F(P_Z, z)$; and finally according to corollary 4.5, $F(P_Z, z)[\![z \cdot \mathtt{id}]\!] \leftrightarrow^*_{\sigma_\in} F(P_Z)$.

We then check that P is convertible to

$$\mathrm{F}_\varepsilon(\bar{\forall}\forall y \forall z(z \in \{z \in y \mid P_Z\} \Leftrightarrow (z \in y \wedge P_Z)))$$

which is the translation of an instance of the comprehension scheme since y does not occur free in P_Z.

The argument is the same if P has been obtained from A_2. □

Lemma 4.38 *Let P be a proposition of Z_b. If $\mathrm{F}_\varepsilon(P)$ is provable in $\in_{es}$ under the axioms A_1, A_2 then P is provable in Z_b under the axioms $\mathcal{S}_1 \cup \mathcal{S}_2$.*

Proof. By hypothesis there exists a list L of propositions of A_1, A_2 such that the sequent $L \vdash \mathrm{F}_\varepsilon(P)$ is provable. According to lemma 4.36, there exists a list M of formulas of $\mathcal{B}$ such that $M \vdash \phi(\mathrm{F}_\varepsilon(P))$ is provable.

All the propositions of M are in $\mathrm{F}_\varepsilon(\mathcal{S}_1 \cup \mathcal{S}_2)$ and $\mathrm{F}_\varepsilon(P)$ is invariant by ϕ. According to theorem 4.24, P is provable under the axioms $\mathcal{S}_1 \cup \mathcal{S}_2$. □

Now, we conclude this section: we have given a translation (F_ε) from the language of Z_b into the language of $\in_{es}/\mathrm{Z}_{es}$ such that any proposition P is provable if and only if $\mathrm{F}_\varepsilon(P)$ is provable in $\in_{es}$. Then we have shown that, in $\in_{es}$, we can express each scheme of Z_b as one axiom of Z_{es}. So, we have that the theory Z_{es} conservatively interprets the theory Z_b. The theory Z_{es} is expressed in deduction modulo but can also be expressed in first-order predicate logic as in section 2; this last presentation being a conservative extension of Z_{es} (see [DHK98] for a proof).

Finite presentations of variants of Z_b We have worked with Z_b but we can give a presentation of any Z_b-based set theory. For instance it is immediate to give a finite presentation that include any finite subset of the individual axioms of Z_b: pairing, null set, sum set, powerset, infinity, regularity, *etc.*

Axiom schemes must be handled one by one since we do not have a formal definition of what a scheme is.

Now, we are going to show how to add the replacement scheme to our presentation. In first-order predicate logic, this scheme is expressed using the following description:

($\mathcal{S}_3$) for any proposition P whose free variables are among $x, y, x_1 \dots x_n$,

$$\begin{aligned}\forall x_1 \dots \forall x_n \forall x \, ((\forall x \forall y \forall z \, ((P \wedge \{z/y\}P) \Rightarrow y = z)) \\ \Rightarrow \exists y \forall u \, (u \in y \Leftrightarrow \exists v (\{v/x; u/y\}P \wedge v \in x)))\end{aligned}$$

The encoding we give is (axiom A_3):

$$\begin{aligned}\forall P \forall x \, ((\forall x \forall y \forall z \, ((\varepsilon(P[\![x \cdot y \cdot \mathtt{id}]\!]) \wedge \varepsilon(P[\![x \cdot z \cdot \mathtt{id}]\!])) \Rightarrow \varepsilon(y \doteq z))) \\ \Rightarrow \exists y \forall u \, (\varepsilon(u \in y) \Leftrightarrow \exists v \, (\varepsilon(P[\![v \cdot u \cdot \mathtt{id}]\!]) \wedge \varepsilon(v \in x))))\end{aligned}$$

It is easy to check that the results of section 4.4 also hold when we both add $\mathcal{S}_3$ to Z_b and A_3 to Z_{es}.

5 Comparison with von Neumann, Bernays and Gödel's set theory

In this section we give an example to show how the encoding of propositions as classes works in the system of von Neumann, Bernays and Gödel. Only the part of the system we are interested in is described; see [Men87] for the complete system.

NBG is expressed in first-order predicate logic, there is no function symbol and only two predicate symbols: $\_\in\_$ and $M(\_)$. For our example we assume that there is also one binary function symbol $<\_,\_>$.

The objects of the theory are *classes* and are denoted by capital italic letters. A class X is also a set whenever $M(X)$ holds. x denotes a class X that is a *set*. Classes that are not sets are called *proper classes*.

The following abbreviations are used (quantification restricted to sets):

$$\forall x\, P \text{ stands for } \forall X\, M(X) \Rightarrow \{X/x\}P$$
$$\exists x\, P \text{ stands for } \exists X\, M(X) \wedge \{X/x\}P.$$

We use the following abbreviations for encoded tuples:

$$<X> \text{ stands for } X$$
$$<X_1, \ldots, X_n, X_{n+1}> \text{ stands for } <<X_1, \ldots, X_n>, X_{n+1}>$$

Now, we give a skolemized version of the axioms of class existence:

(B1) $\forall u \forall v\ (<u,v> \in E \Leftrightarrow u \in v)$
(B2) $\forall u\ (u \in (X \cap Y) \Leftrightarrow (u \in X \wedge u \in Y))$
(B3) $\forall u\ (u \in -X \Leftrightarrow u \notin X)$
(B4) $\forall u\ (u \in D(X) \Leftrightarrow (\exists v\ <u,v> \in X))$
(B5) $\forall u \forall v\ (<u,v> \in +(X) \Leftrightarrow u \in X)$
(B6) $\forall u \forall v \forall w\ (<u,v,w> \in R(X) \Leftrightarrow <v,w,u> \in X)$
(B7) $\forall u \forall v \forall w\ (<u,v,w> \in S(X) \Leftrightarrow <u,w,v> \in X)$

In the two systems $\in_{es}$ and NBG we are now giving an encoding of:

"the class of the x such that $\exists y((\exists z\ x \in z) \wedge (\exists z\ z \in y))$".

In NBG the encoding is the term: $D(D(S(+(E))) \cap D(S(R(+(E)))))$. Let a be a set; using only axioms B1–B7 we can prove that $a \in D(D(S(+(E))) \cap D(S(R(+(E)))))$ is equivalent to $\exists y((\exists z\ a \in z) \wedge (\exists z\ z \in y))$; the details are given below:

$$\begin{array}{ll} & a \in D(D(S(+(E))) \cap D(S(R(+(E))))) \\ \Leftrightarrow & \exists y\ (<a,y> \in D(S(+(E))) \wedge <a,y> \in D(S(R(+(E))))) \\ \Leftrightarrow & \exists y\ ((\exists z\ <a,y,z> \in S(+(E))) \wedge (\exists z\ <a,y,z> \in S(R(+(E))))) \\ \Leftrightarrow & \exists y\ ((\exists z\ <a,z,y> \in +(E)) \wedge (\exists z\ <a,z,y> \in R(+(E)))) \\ \Leftrightarrow & \exists y\ ((\exists z\ <a,z> \in E) \wedge (\exists z\ <z,y,a> \in +(E))) \\ \Leftrightarrow & \exists y\ ((\exists z\ x \in z) \wedge (\exists z\ <z,y> \in E)) \\ \Leftrightarrow & \exists y\ ((\exists z\ a \in z) \wedge (\exists z\ z \in y)) \end{array}$$

In $\in_{es}$ the encoding is the following term of sort o:

$$\dot\exists(\dot\exists(\mathtt{3} \mathrel{\dot\in} \mathtt{1}) \mathrel{\dot\wedge} \dot\exists(\mathtt{1} \mathrel{\dot\in} \mathtt{2}))$$

Then we check that, for any term a of sort *set*, the proposition

$$\varepsilon((\dot\exists(\dot\exists(\mathtt{3} \mathrel{\dot\in} \mathtt{1}) \mathrel{\dot\wedge} \dot\exists(\mathtt{1} \mathrel{\dot\in} \mathtt{2})))[\![a \cdot \mathtt{id}]\!])$$

reduces to

$$\exists y((\exists z\ \varepsilon(a \mathrel{\dot\in} z)) \wedge (\exists z\ \varepsilon(z \mathrel{\dot\in} y)))$$

As we can see the encoding we provide is more straightforward. Moreover the axiom we have used are already oriented, which is an important point concerning automated reasonning.

Conclusion

We have given a finite first-order presentation of set theory whose syntax is very close to Z_b. We have shown how to obtain this presentation from Z_b and proved that it conservatively interprets Z_b, the proof being done independently of the axioms used in the theory. Therefore the method holds for variants of Z_b.

The first-order presentation we have given can be expressed in deduction modulo where the part of the theory dealing with the encoding is expressed as a congruence. We have used this presentation as an intermediate system for the proof of equivalence but it may also be a suitable presentation for proof search since the search space is restricted. From this point of view it seems to be worth trying to convert some of the axioms of set theory into rewrite rules, for instance the translation of the powerset axiom can be expressed as the rule: $x \dot{\in} \mathcal{P}(y) \to \dot{\forall}(1 \dot{\in} x[\uparrow] \Rightarrow 1 \dot{\in} y[\uparrow])$. For automated theorem proving it is important to try to have a set of rewrite rules as small as possible, therefore it might be worth using another calculus of explicit substitution in place of $\lambda\sigma_{\Uparrow}$ but we have to keep in mind that this calculus must enjoy confluence on terms with term meta-variables.

The encoding of NBG is not as straightforward as ours: in $\in_{\Uparrow}$ the structure of propositions coded as terms is preserved whereas it is not the case for NBG; this has been the consequence of using an explicit substitution with de Bruijn indices. Nonetheless section 5 shows similarities between the two systems and it would be interesting to do a detailed comparison.

We do not propose a general method for encoding schemes as a single axiom. For instance it doesn't seem to be possible, without modification of our system, to encode the reflection scheme [Kri98] in which occur both a proposition P and P where the quantifiers are restricted to a class Y (*e.g.* $\forall x \ldots$ becomes $\forall x(Y(x) \Rightarrow \ldots)$).

Acknowledgments

The author wishes to thank Pierre-Louis Curien, Gilles Dowek, Thérèse Hardin and anonymous referees.

References

[Bar84] Hendrik Pieter Barendregt. *The Lambda Calculus – Its Syntax and Semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, revised edition, 1984.

[BLM+86] Robert Boyer, Ewing Lusk, William McCune, Ross Overbeek, Mark Stickel, and Lawrence Wos. Set theory in first-order logic: clauses for Gödel's axioms. *Journal of Automated Reasoning*, 2:287–327, 1986.

[CHL96] Pierre-Louis Curien, Thérèse Hardin, and Jean-Jacques Lévy. Confluence properties of weak and strong calculi of explicit substitutions. *Journal of the ACM*, 43(2):362–397, March 1996.

[DHK98] Gilles Dowek, Thérèse Hardin, and Claude Kirchner. Theorem proving modulo. Technical report, INRIA, April 1998. Rapport de recherche 3400.

[DHK99] Gilles Dowek, Thérèse Hardin, and Claude Kirchner. HOL-$\lambda\sigma$: An intentional first-order expression of higher-order logic. In P. Narendran and M. Rusinowitch, editors, *Proceedings of the 10th International Conference on Rewriting Techniques and Applications (RTA-99)*, pages 317–331, Trento, Italy, July 1999. Springer-Verlag LNCS 1631. Rapport de Recherche 3556, INRIA 1998. Mathematical Structures in Computer Science 11 (2001) pp. 1-25.

[DHK00] Gilles Dowek, Thérèse Hardin, and Claude Kirchner. Higher-order unification via explicit substitutions. *Information and Computation*, pages 183–235, 2000. Logic in Computer Science, pp.366–374, 1995. Rapport de Recherche 2709, INRIA 1995.

[Dow95] Gilles Dowek. Lambda-calculus, combinators and the comprehension scheme. In *Typed Lambda Calculi and Applications*, pages 154–170, 1995. Rapport de Recherche 2565, INRIA 1995.

[DW99] Gilles Dowek and Benjamin Werner. Proof normalization modulo. In *Lecture Notes in Computer Science 1657*, pages 62–77, 1999. Rapport de Recherche 3542, INRIA 1998.

[GLT89] Jean-Yves Girard, Yves Lafont, and Paul Taylor. *Proofs and Types*. Cambridge, 1989.

[Hin64] J. Roger Hindley. *The Church-Rosser Property and a Result in Combinatory Logic*. PhD thesis, University of Newcastle-upon-Tyne, 1964.

[Kri98] Jean-Louis Krivine. *Théorie des ensembles*. Cassini, 1998.

[KvOvR93] J.W. Klop, V. van Oostrom, and F. van Raamsdonk. Combinatory reduction systems: introduction and survey. *Theorical Computer Science*, 121:279–308, 1993.

[Men87] Elliott Mendelson. *Introduction to Mathematical Logic*. Chapman & Hall, Third edition, 1987.

[Pag98] Bruno Pagano. X.R.S : eXplicit Reduction Systems, a first-order calculus for higher-order calculi. In *Conference on Automated Deduction, Lindau*, pages 66–80, July 1998.

[Sup72] Patrick Suppes. *Axiomatic Set Theory*. Dover, 1972.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399