Hilbert's ϵ-Terms in Automated Theorem Proving

Martin Giese and Wolfgang Ahrendt

Institut für Logik, Komplexität und Deduktionssysteme,
Universtät Karlsruhe, Germany
{giese|ahrendt}@ira.uka.de

Abstract. ϵ-terms, introduced by David Hilbert [8], have the form $\epsilon x.\phi$, where x is a variable and ϕ is a formula. Their syntactical structure is thus similar to that of a quantified formulae, but they are terms, denoting 'an element for which ϕ holds, if there is any'.

The topic of this paper is an investigation into the possibilities and limits of using ϵ-terms for automated theorem proving. We discuss the relationship between ϵ-terms and Skolem terms (which both can be used alternatively for the purpose of $\exists$-quantifier elimination), in particular with respect to efficiency and intuition. We also discuss the consequences of allowing ϵ-terms in theorems (and cuts). This leads to a distinction between (essentially two) semantics and corresponding calculi, one enabling efficient automated proof search, and the other one requiring human guidance but enabling a very intuitive (i.e. semantic) treatment of ϵ-terms. We give a theoretical foundation of the usage of both variants in a single framework. Finally, we argue that these two approaches to ϵ are just the extremes of a range of ϵ-treatments, corresponding to a range of different possible Skolemization variants.

1 Introduction

Calculi for full first-order predicate logic have to cope with the elimination of existential quantifiers. Quantified variables are usually replaced by terms, which have to obey certain restrictions. Many approaches in proof theory and almost all approaches in automated deduction use the concept of Skolem functions (resp. constants) for this purpose. An alternative concept for terms replacing existentially quantified variables is that of ϵ-terms. An ϵ-term has the form $\epsilon x.\phi$, where x is a variable and ϕ is a formula. The intended meaning is 'an element for which ϕ holds, if there is any, and an arbitrary element otherwise'. If ϕ holds for more than one element, or for none, ϵ acts as a choice operator.

A Skolem term introduced during elimination of the quantifier in $\exists x.\phi(x)$ also denotes an element e for which $\phi(e)$ holds. But in contrast to the Skolem term, the ϵ-term refers explicitly (on an object language level) to the property ϕ it satisfies.

1.1 Short History of ϵ-Terms

The ϵ-symbol was introduced by Hilbert in the context of the formalist effort to prove the consistency of arithmetic and analysis by finitary means. In particular, ϵ-terms are used to give a finitary justification of the use of (non-finitary) quantifier reasoning in predicate logic. The arguments in this context are typically based on proof transformations. Model-theoretic reasoning would have been inappropriate, as reasoning about models is usually non-finitary. The principal work in this area is by Hilbert and Bernays [8]. Leisenring [9] gives a more condensed and up-to-date survey of the field.

In the context of *automated deduction,* reasoning with models is not regarded as problematic. Indeed, soundness or completeness statements are almost always relative to a given model semantics. Possible model semantics for ϵ-terms are investigated by Meyer Viol [10] and also to a certain degree by Leisenring.

To our knowledge, in the context of automated deduction, calculi do not use ϵ-terms as a syntactical construct. On the other hand, the development of improved δ-rule versions (see below) can be seen as a progressing approximation of ϵ-like behaviour.

1.2 Short History of δ-Rules

Elimination of existential quantifiers takes place either in a preprocessing step or, in particular in analytic non-normal form calculi (i.e. tableaux and sequent calculi), in a special expansion rule, called δ-rule. The evolution of different δ-rules that we sketch now took place in the framework of tableaux. We use the tableau notation in the rest of this paper.[1]

In a Smullyan style *ground* tableau calculus [11], there is a δ-rule of the form

$$\frac{\exists x.\phi(x)}{\phi(c)} \ ,$$

where c is a constant symbol, which must be new relative to the tableau or branch to which the rule is applied. The intuition behind this requirement is to make sure that all we know about c is $\phi(c)$. (Sometimes, we *do* know more about c, however, which is where some liberalized δ-rules come in.)

In a *free variable* tableau calculus, where free variables stand for instances not yet known, the δ-rule

$$\frac{\exists x.\phi(x)}{\phi(f(x_1,\ldots,x_n))}$$

introduces a term $t = f(x_1, \ldots, x_n)$, where the choice of both the function symbol and the variables has to meet certain requirements, which vary from one δ-rule to the other. Early versions of this rule, e.g. [5], required that the function symbol f is new and that all free variables present on the current branch are

[1] Note, however, that it is trivial to translate the discussion to a sequent calculus notation.

parameters of t. These parameters guarantee that t stays new w.r.t. the branch even after applying arbitrary substitutions.

Later versions of the δ-rule for free variable tableaux modified the restrictions for t, always shortening the minimal proof length. At first, the δ^+-rule, introduced by Hähnle and Schmitt [7], reduced the parameters of t to the free variables of the expanded formula only. Now, t is actually a *Skolem term* in the sense, that the soundness argument for this rule uses the semantic properties of Skolemization. Note that, with this rule, it is possible to unify t with a free variable occurring above in the branch. Consequently, after applying such a unifier to the tableau, the term replacing the existentially bound variable occurs in the proof prior to the rule that introduced it. It is not trivial to formulate a sound δ-rule for a *ground* tableau, which corresponds one to one to δ^+, see [2, Sect. 3.6].

A further modification of restrictions for t is formulated in the δ^{++}-rule by Beckert et. al.[3]. Now, the function symbol of the Skolem term need not be new in general. Instead, the same functor can be used when the δ^{++}-rule is applied to formulae that are identical up to renaming of (free and bound) variables. In theory, classes of such formulae *are* the functors. This way of Skolemization is closely related to the idea of ϵ-terms, because the chosen element is identified by a class of formulae it satisfies. However, Skolemization of two formulae, where one is an instance of the other, leads to non-unifiable results. Consequently, δ^{++}-rule application and substitution of free variables are not exchangeable, which is unsatisfying from an intuitive point of view.

There are already δ-rules going beyond δ^{++}, e.g. δ^* [1] and δ^{*^*} [4], which we shall come back to in the course of this paper.

1.3 This Paper

This paper is concerned with the embedding of ϵ-terms in a calculus well suited for automated theorem proving. Moreover, our issue is the border between ϵ-handling that fits purely automated proof search, and ϵ-handling requiring human guidance. The context of our work is research on concepts for integrating automated and interactive theorem proving in a *homogeneous* way. By 'homogeneous' we mean an integration of the two paradigms in *one* prover, based on *one* calculus. In this setting, a calculus must be intuitive, as well as efficient, which shall be an issue in Sect. 3 and 4.

In this paper, we present a spectrum of treatments of ϵ-terms, discussing their suitability for automated proof search. ϵ essentially is a choice operator. Therefore, fixing its semantics means fixing the features of the choice. Given $\exists x.\phi(x)$, the choice of an element e, for which $\phi(e)$ holds, may for example depend only on the semantics (i.e. the extension) of ϕ. Another possibility is to let the choice depend only on the syntax of a formula (compare δ^{++} above). But then, the choice function should have some basic properties, which we discuss below.

We start, in Sect. 2, with the introduction of a δ^ϵ-rule and, because of the similarity to the δ^{++}-rule, compare both with respect to minimal proof length.

Then, we turn to the semantics of ϵ-terms in Sect. 3, defining a hierarchy of ϵ-structures. The distinction between different structures is justified in Sect. 4, where two calculi that are complete for different semantics are presented and discussed with respect to automated theorem proving.

2 Using ϵ-Terms Instead of Skolem Functions

2.1 Introducing ϵ-Terms

We begin by defining a number of basic syntactic notions.

Definition 1 (Syntax, free/bound variables, substitutions). *Let* $\mathbf{V}$ *be a fixed (infinite) set of* **variables**. *The sets* $\mathbf{Tm}$*, resp.* $\mathbf{Fm}$*, of well formed first order* **terms***, resp.* **formulae***, are defined as usual, with the additional requirement, that for all* $x \in \mathbf{V}$ *and* $\phi \in \mathbf{Fm}$*, there is a term* $\epsilon x.\phi \in \mathbf{Tm}$.[2]

For a term or formula α*, define* $\mathrm{bv}(\alpha) \subseteq \mathbf{V}$*, resp.* $\mathrm{fv}(\alpha) \subseteq \mathbf{V}$*, the sets of* **bound***, resp.* **free variables** *of* α*. A term, resp. formula is called* **closed** *if it has no free variables. The sets of all closed terms, resp. closed formulae are denoted by* $\mathbf{Tm}^0$*, resp.* $\mathbf{Fm}^0$.

A **substitution** *is a mapping* $\sigma : \mathbf{V} \to \mathbf{Tm}$*, where* $\mathrm{dom}(\sigma) := \{x \in \mathbf{V} \mid \sigma(x) \neq x\}$ *(called the* **domain** *of* σ*) is finite. The notation* $\sigma = [x_1/t_1, \ldots, x_n/t_n]$ *is used for the substitution with* $\sigma(x_i) = t_i$*,* $\mathrm{dom}(\sigma) = \{x_1, \ldots, x_n\}$.

The most important point here is that terms may contain bound variables, which is not the case in ordinary first order logic: $\epsilon x.\phi$ is a term in which the variable x is bound. This means that a little more care needs to be taken, when arguing about substitutions.

Instead of giving a formal semantics for ϵ-terms right away, we first show what we want to use them for, and defer the rigorous discussion to Sect. 3. The given intuition behind ϵ-terms captures the essence of $\exists$-quantifier elimination: given $\exists x.\phi$, $\epsilon x.\phi$ denotes a value of which we know nothing, except that it makes ϕ true. Accordingly, we use the δ-rule

$$\frac{\exists x.\phi(x)}{\phi(\epsilon x.\phi(x))}\ \delta^{\epsilon} \quad .$$

To give the reader a general idea of how this works, here is a proof of the inconsistency of the set of formulae

$$\{\forall u.p(u,a,b), (\forall y.\exists x.\neg p(x,y,b)) \vee (\forall z.\exists x.\neg p(x,a,z))\}$$

[2] This means, that *unlike* the usual practice, terms and formulae are defined by *mutual* recursion.

in an unsigned tableau-calculus with free variables:

$$
\begin{array}{c}
1: \ \forall u.p(u,a,b) \\
2: \ (\forall y.\exists x.\neg p(x,y,b)) \lor (\forall z.\exists x.\neg p(x,a,z)) \\
3, \gamma(1): \ p(U,a,b)
\end{array}
$$

$$
\begin{array}{cc}
4, \beta(2): \ \forall y.\exists x.\neg p(x,y,b) & 5, \beta(2): \ \forall z.\exists x.\neg p(x,a,z) \\
6, \gamma(4): \ \exists x.\neg p(x,Y,b) & 7, \gamma(5): \ \exists x.\neg p(x,a,Z) \\
8, \delta^{\epsilon}(6): \ \neg p(\epsilon x.\neg p(x,Y,b),Y,b) & 9, \delta^{\epsilon}(7): \ \neg p(\epsilon x.\neg p(x,a,Z),a,Z)
\end{array}
$$

The tableau is closed after applying the following substitution:

$$[U/\epsilon x.\neg p(x,a,b), Y/a, Z/b]$$

With the δ, δ^{+}, δ^{++} or δ^{*} rules, different skolem functions would be chosen for the skolemization of formulae 6 and 7, so the tableau could not be closed without a second instance of the γ-formula 1.

It should be mentioned at this point, that ϵ-terms (a) may be nested, (b) may contain free variables, and (c) may lead to rather large formulae, as they repeat most of the δ-formula. The problem of large formulae can be addressed in an implementation using structure sharing.

The main benefit of using ϵ-terms to handle δ-formulae is that identical formulae lead to introduction of the same term. The same idea is realized in the δ^{++}-rule. Therefore, in the next section we compare that rule to δ^{ϵ}.

2.2 Exponentially Shorter Proofs with ϵ-terms than with δ^{++}

We shall now show that the δ^{ϵ}-rule can cut down minimal proof-length exponentially with respect to a certain modification of the δ^{++}-rule: while the original δ^{++}-rule allows to assign the same Skolem-function symbol to any two formulae which are equal up to renaming of *bound and free* variables, we require the formulae to be equal up to renaming of *free variables only.* We refer to this modification as the $\delta^{+^{-}}$-rule.

Theorem 1 (Proof length with δ^{ϵ} vs. $\delta^{+^{-}}$). *There is a family $\phi_n, n \in \mathbb{N}$ of valid first order formulae, such that the minimal number b^{ϵ}, resp. $b^{+^{-}}$ of branches in a closed tableau for ϕ_n with the δ^{ϵ}-rule, resp. $\delta^{+^{-}}$-rule satisfy $b^{\epsilon}(n) \in \Theta(n)$, and $b^{+^{-}}(n) \in \Theta(2^n)$.*

Proof. The proof is based on the same ideas as the one in [3], where it is shown, that the δ^{++}-rule permits exponentially shorter proofs than the δ^{+}-rule. Define

$$
\begin{aligned}
\phi_0 &:= \mathit{true} \\
\phi_{n+1} &:= \exists x.\Big(\phi_n \land \big(p_n(x,a,b) \to (\exists y.\forall x.p_n(x,y,b) \land \exists z.\forall x.p_n(x,a,z))\big)\Big) \\
& \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \text{for } n \in \mathbb{N}.
\end{aligned}
$$

The proof proceeds analogous to that of [3]. As in the introductory example of section 2.1, the inclusion of the skolemized formula in the ϵ-terms provides the necessary information to permit the simultaneous closure of two branches in the δ^{ϵ} case, where another γ-rule application is needed with $\delta^{+^{-}}$. □

Clearly, any $\delta^{+^{-}}$-proof can be simulated using δ^{ϵ}, so the δ^{ϵ}-rule is strictly stronger than $\delta^{+^{-}}$.

Remark 1. It is not hard to modify the δ^{ϵ}-rule to obtain exponentially shorter minimal proofs than with the origninal δ^{++}-rule of [3]: one only needs to define closure by means of unification *modulo* renaming of bound variables. Alternatively, normalize the names of bound variables when applying the δ^{ϵ}-rule. We will omit this technical detail here, however.

Remark 2. Baaz and Fermüller [1] show a stronger speed-up result, namely that the δ^{++} rule gives *non-elementary* speedup w.r.t. to the δ^{+} rule. We are currently investigating whether their proof technique can be applied to show that δ^{ϵ} yields non-elementary speed-up w.r.t. δ^{++}.

Remark 3. It is also possible to strengthen the δ^{ϵ}-calculus in a way that makes it *strictly stronger* than the δ^{*}-rule of Baaz and Fermüller [1], which in turn gives non-elementary speed-up w.r.t. the δ^{++}-rule. For lack of space, we are not going to develop this any further in this paper.

3 Semantics of ϵ-Terms

In the last section, we have introduced ϵ-terms as syntactical entities, but we have not given them a formal model-semantics, which is the topic of this section.

3.1 Valuation in Pre-Structures

We want our logic with ϵ-terms to be a conservative extension of classical predicate logic, i.e. the validity of terms and formulae that do not contain ϵ-terms should remain the same. Accordingly, the valuation functions correspond closely to the classical case. On the other hand, we will discuss several possible semantics for ϵ-terms, so we give some minimal semantic definitions first and refine them later.

Definition 2 (Variable assignments, pre-structures). *A* **variable assignment** *of* $\mathbf{V}$ *to a set* $\mathcal{D}$ *is a function* $\beta : \mathbf{V} \to \mathcal{D}$*. We denote by* $\beta\{x \leftarrow d\}$ *the modified assignment with*

$$\beta\{x \leftarrow d\}(y) := \begin{cases} d & \textit{if } y = x, \\ \beta(y) & \textit{otherwise.} \end{cases}$$

A **pre-structure** *is a triple* $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ *with the following properties:*

- $(\mathcal{D}, \mathcal{I})$ *is a classical first order structure with* **carrier** $\mathcal{D}$ *and* **interpretation** $\mathcal{I}$*.*

– *The* ϵ-**valuation** $\mathcal{A}$ *is a function that maps any* ϵ*-term* $\epsilon x.\phi$ *and any variable assignment* β *on* $\mathcal{D}$ *to a value* $\mathcal{A}(\epsilon x.\phi, \beta) \in \mathcal{D}$.

This definition contains no restriction whatsoever on the valuation of ϵ-terms. We will add restrictions that reflect the intended behaviour of these terms later. Here, we proceed by defining the valuation of terms and formulae on pre-structures.

Definition 3 (Term and formula valuation). *The* **valuation** $\mathrm{val}(\mathcal{S}, \beta, t) \in \mathcal{D}$ *of a term* $t \in \mathbf{Tm}$ *in a pre-structure* $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ *under a variable assignment* β *is defined as for classical first order logic, except for the valuation of* ϵ*-terms, where we set*

$$\mathrm{val}(\mathcal{S}, \beta, \epsilon x.\phi) := \mathcal{A}(\epsilon x.\phi, \beta) \ .$$

The validity relation for formulae, $\mathcal{S}, \beta \models \phi$ *is defined exactly as for classical first order logic.*

Note, that – in contrast to the syntax – *no* mutual recursion between terms and formulae is needed in these semantic definitions: the whole valuation of ϵ-terms is delegated to the function $\mathcal{A}$, so the semantic definitions do not take the formula in an ϵ-term into account so far.

3.2 A Hierarchy of Structures

In this section, we give several concrete restrictions leading to more useful semantics for ϵ-terms. In particular, we define the *substitutive* and *extensional* semantics, for which we give complete calculi in Sect. 4.

Two minimal requirements are needed to ensure a sensible semantics for ϵ-terms: first, the valuation of an ϵ-term should depend only on the valuation of variables occurring free in that term. Second, an ϵ-term $\epsilon x.\phi$ should actually denote a value that satisfies ϕ, if any such value exists. These requirements are captured in the following definition:

Definition 4 (Intensional structure). *A pre-structure* $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ *is called* **intensional structure** *or* **I-structure***, if*

– *any* ϵ*-term* $\epsilon x.\phi$ *and two assignments* β_1, β_2 *with* $\beta_1|_{\mathrm{fv}(\epsilon x.\phi)} = \beta_2|_{\mathrm{fv}(\epsilon x.\phi)}$ *satisfy* $\mathcal{A}(\epsilon x.\phi, \beta_1) = \mathcal{A}(\epsilon x.\phi, \beta_2)$.
– *for any* β, $x \in \mathbf{V}$, $\phi \in \mathbf{Fm}$, *if* $\mathcal{S}, \beta \models \exists x.\phi$, *then* $\mathcal{S}, \beta\{x \leftarrow \mathcal{A}(\epsilon x.\phi, \beta)\} \models \phi$.

A formula which is valid in all I-structures under all variable assignments is called **I-valid***. If it is valid in at least one I-structure under at least one variable assignment, it is called* **I-satisfiable***.*

This intensional semantics lacks an important property: it is not *substitutive*. E.g., from $\forall x.q(\epsilon y.p(x, y))$ it is not possible to infer $q(\epsilon y.p(a, y))$. Similarly, from the equality $a = b$, we can not infer $\epsilon x.p(x, a) = \epsilon x.p(x, b)$. However, these inferences become possible, if we further constrain the set of permissible structures.

Definition 5 (Substitutive structure). *An I-structure* $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ *is called* **substitutive** *or* **S-structure***, if for all* $x, y \in \mathbf{V}$*,* $\phi \in \mathbf{Fm}$*,* $\beta : \mathbf{V} \to \mathcal{D}$ *and* $t \in \mathbf{Tm}$ *with* $\mathrm{fv}(t) \cap \mathrm{bv}(\epsilon x.\phi) = \emptyset$*,*

$$\mathcal{A}([y/t](\epsilon x.\phi), \beta) = \mathcal{A}(\epsilon x.\phi, \beta\{y \leftarrow \mathrm{val}(\mathcal{S}, \beta, t)\}) \enspace .$$

S-validity *and* **S-satisfiability** *are defined analogous to Definition 4.*

Substitutivity, namely the fact that

$$\mathrm{val}(\mathcal{S}, \beta, [y/t]\alpha) = \mathrm{val}(\mathcal{S}, \beta\{y \leftarrow \mathrm{val}(\mathcal{S}, \beta, t)\}, \alpha)$$

for any term or formula α with $\mathrm{fv}(\alpha) \cap \mathrm{bv}(\epsilon x.\phi) = \emptyset$, follows directly from this definition for S-structures. Substitutivity is a central property for the construction of a calculus, as it captures the semantic effects of the syntactic operation of substituting parts of a term or formula.

Classical first order logic has the property that replacing an arbitrary subformula ψ of a formula ϕ by a logically equivalent formula ψ' maintains the validity of ϕ. This is not necessarily the case with S-validity. In fact, from $\forall x.p(x) \leftrightarrow q(x)$ it does not follow that $\epsilon x.p(x) = \epsilon x.q(x)$. As long as we use ϵ-terms for $\exists$-quantifier elimination only, this would not be a problem. But, as we argue in the next section, it is reasonalbe to permit the use of ϵ-terms in the formulation of problems, which might well be done by a human. In that case, it is vital to make the behaviour of ϵ-terms as intuitive as possible. The main intuition behind logical equivalence is that replacing part of a formula by something equivalent should not change the meaning of the whole. We therefore define a semantics that has this property, by making the interpretation of an ϵ-term $\epsilon x.\phi$ depend on the *semantics* of the formula ϕ.

Definition 6 (Extensional structure). *For an* ϵ*-term* $\epsilon x.\phi$*, an I-structure* $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ *and a variable assignment* $\beta : \mathbf{V} \to \mathcal{D}$*, define the* **extension**

$$\mathrm{Ext}(\mathcal{S}, \beta, \epsilon x.\phi) := \{d \in \mathcal{D} \mid \mathcal{S}, \beta\{x \leftarrow d\} \models \phi\}$$

An I-structure is called **extensional** *or* **E-structure***, if for all* $x, y \in \mathbf{V}$*,* $\phi, \psi \in \mathbf{Fm}$*,* $\beta : \mathbf{V} \to \mathcal{D}$*,*

$$\textit{if}\ \ \mathrm{Ext}(\mathcal{S}, \beta, \epsilon x.\phi) = \mathrm{Ext}(\mathcal{S}, \beta, \epsilon y.\psi), \ \ \textit{then}\ \ \mathcal{A}(\epsilon x.\phi, \beta) = \mathcal{A}(\epsilon y.\psi, \beta) \enspace .$$

E-validity *and* **E-satisfiability** *are defined analogous to Definition 4.*

The three variations of ϵ-term semantics constitute a hierarchy, as stated in the following theorem.

Theorem 2 (Hierarchy Theorem). *Let* $\phi \in \mathbf{Fm}$*. If* ϕ *is E-satisfiable, then it is S-satisfiable. If* ϕ *is S-satisfiable, then it is I-satisfiable.*

Proof. The only non-trivial part of the proof is to show that extensional structures are always substitutive, which is done by showing substitutivity for all formulae and terms, using structural induction. For the complete proof, see [6] or [9]. ☐

It was mentioned at the beginning of this section, that the logic with ϵ-terms should be a conservative extension of classical first order logic, whatever the exact semantics chosen for the ϵ-terms. This is ensured by the following theorem.[3]

Theorem 3 (Embedding Theorem). *Let $\phi \in \mathbf{Fm}$ be a formula* without ϵ-terms. *The following statements are equivalent:*

1. *ϕ is satisfiable in classical first order logic.*[4]
2. *ϕ is I-satisfiable.*
3. *ϕ is S-satisfiable.*
4. *ϕ is E-satisfiable.*

Proof. **1.⇒4.:** Let $\mathcal{S}_0 = (\mathcal{D}, \mathcal{I})$ be a classical first order structure, and $\beta : \mathbf{V} \to \mathcal{D}$ a variable assignment such that $\mathcal{S}_0, \beta \models \phi$. We show the existence of an E-structure $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ with $\mathcal{S}, \beta \models \phi$. As ϕ does not contain ϵ-terms, the validity of ϕ does not depend on $\mathcal{A}$. Thus, it suffices to construct *any* E-structure with carrier $\mathcal{D}$ and interpretation $\mathcal{I}$.

Using the axiom of choice, we may assume the existence of a function $\alpha : \mathcal{P}(\mathcal{D}) \to \mathcal{D}$ satisfying $\alpha(M) \in M$ for all non-empty sets $M \subseteq \mathcal{D}$. The ϵ-valuation $\mathcal{A}$ is defined by successive approximation. We define the family of sets $F_i \subset \mathbf{Fm}$ for $i \in \mathbb{N}$ by:

$$\begin{aligned} F_0 &:= \{\phi \in \mathbf{Fm} \mid \phi \text{ contains no } \epsilon\text{-terms}\} \\ F_{i+1} &:= F_i \cup \{\phi \in \mathbf{Fm} \mid \phi \text{ contains only } \epsilon\text{-terms } \epsilon x.\phi' \text{ with } \phi' \in F_i\} \end{aligned}$$

Obviously, we have $\mathbf{Fm} = \bigcup_{i=0}^{\infty} F_i$. Let $\iota(\phi) := \min\{i \mid \phi \in F_i\}$ be the first of these sets containing a given formula ϕ. We now define a family of ϵ-valuations $\mathcal{A}_i$ for $i \in \mathbb{N}$ as follows:

$$\begin{aligned} \mathcal{A}_0(\epsilon x.\phi, \beta) &:= d_\perp \\ \mathcal{A}_{i+1}(\epsilon x.\phi, \beta) &:= \begin{cases} \alpha(\mathrm{Ext}_{\iota(\phi)}(\epsilon x.\phi)) & \text{for } \phi \in F_i, \\ d_\perp & \text{otherwise,} \end{cases} \end{aligned}$$

where $\mathrm{Ext}_i(\epsilon x.\phi) := \{d \in \mathcal{D} \mid (\mathcal{D}, \mathcal{I}, \mathcal{A}_i), \beta\{x \leftarrow d\} \models \phi\}$, and $d_\perp \in \mathcal{D}$ is an arbitrary carrier element. Defining

$$\mathcal{A}(\epsilon x.\phi, \beta) := \mathcal{A}_{\iota(\phi)+1}(\epsilon x.\phi, \beta)$$

makes $\mathcal{S} := (\mathcal{D}, \mathcal{I}, \mathcal{A})$ an E-structure. The proof that this is the case is not very hard, though somewhat technical, and can be found in [6].

4.⇒3. and **3.⇒2.** follow immediately from Theorem 2.

2.⇒1.: The validity of ϕ is independent of the ϵ-valuation, as ϕ contains no ϵ-terms. Therefore, $(\mathcal{D}, \mathcal{I}, \mathcal{A}), \beta \models \phi$ implies $(\mathcal{D}, \mathcal{I}), \beta \models \phi$ in classical first order logic. □

[3] This is the semantic equivalent of Hilbert's Second ϵ-Theorem. It is of course much easier to show, because we argue with model-semantics instead of proof theory.

[4] The definition of satisfiability differs slightly between authors. We call a formula satisfiable if there are a structure and a variable assignment which satisfy the formula.

It should be remarked, that there are many more variants of ϵ-semantics than the three proposed in this paper. The intensional semantics is minimal, in the sense, that it captures only the most basic properties of ϵ-terms. The extensional semantics, on the other hand, assures an intuitive structural property. Finally, as the next section shows, the substitutive semantics has pleasant properties when it comes to constructing a calculus. But there are of course many other possible restrictions on the evaluation of ϵ-terms, that give rise to as many different semantics. E.g., it is possible to require the value of ϵ-terms to remain the same under renaming of bound variables, a property that is guaranteed in E-structures, but not in S-structures. That would permit a full simulation of the δ^{++}-rule. It is also possible to construct an even stronger semantics than the extensional one: for instance, one might require the existence of a well-ordering on the carrier set $\mathcal{D}$, such that the value of an ϵ-term is always the *minimal* element of its extension. In view of the results of the next section, however, stronger semantics are probably not of much interest to automated theorem proving.

4 Proving theorems with ϵ-Terms

If we restricted the use of ϵ-terms to $\exists$-quantifier elimination, the completeness of the resulting calculus for first order problems – without ϵ-terms – would be an easy consequence of the completeness of less liberal δ-rules, like δ^+ or δ^{++}. The main thing to show would be the soundness of the new rule.

However, the work presented in this paper was done with the aim of integrating automated and interactive proof systems using a common calculus. In that setting, it seemed unnatural to forbid the use of ϵ-terms in the formulation of the proof obligations themselves. The user might want to formulate lemmata or cut-formulae that use ϵ-terms. So the question was, whether we could find a calculus that was complete for the whole logic with ϵ-terms, or more precisely, for which semantics such a calculus could be found.

We now present variants of the free-variable tableau calculus for the substitutive and extensional semantics; the intensional semantics, lacking substitutivity, is to weak to allow a reasonable free-variable calculus. The calculus for the extensional semantics will use a logic with equality, but we shall not discuss equality *handling* here, as the problems arising are largely orthogonal. For a more detailed discussion, including equality handling with constraints, see [6].

4.1 A Complete Calculus for the Substitutive Semantics

We consider a standard unsigned free-variable tableau calculus with the usual α, β and γ expansion rules, as well as a closure rule based on syntactic unification, that applies a substitution to all formulae in the tableau. We use the following δ^ϵ expansion rules:

$$\frac{\exists x.\phi}{[x/\epsilon x.\phi]\phi} \quad \text{and} \quad \frac{\neg\forall x.\phi}{[x/\epsilon x.\neg\phi]\neg\phi} \ .$$

Additionally we introduce an ϵ expansion rule,

$$\frac{}{\forall x.\neg\phi \quad | \quad [x/\epsilon x.\phi]\phi.}$$

In the left branch, one has to show, that there exists at least one element satisfying ϕ. In the right branch, we can use the fact that $\epsilon x.\phi$ denotes one such element.

By taking $x \notin \mathrm{fv}(\phi)$, this rule can easily be seen to be equivalent to the *cut*-rule! So, to permit the application of the ϵ-rule in an automated theorem prover without exploding the search space, we have to make sure that it is only applied in a very limited way. We show that the calculus remains complete, if we allow the application of the ϵ-rule only if

1. the branch contains an atomic formula $(\neg)p(t_1, \ldots, t_n)$, such that
   (a) $\epsilon x.\phi$ is a subterm of one of the terms t_i,
   (b) no free variable of $\epsilon x.\phi$ is bound by a containing ϵ-term in t_i,
2. $\epsilon x.\phi$ was not introduced by a δ-rule, and
3. the ϵ-rule has not previously been applied for $\epsilon x.\phi$ on this branch.

For instance, given an atom

$$p(f(\epsilon x.q(x,y)), \epsilon x.r(g(\epsilon y.s(x,y)))) \ ,$$

the ϵ-rule is applied for $\epsilon x.q(x,y)$ and $\epsilon x.r(g(\epsilon y.s(x,y)))$, but not for $\epsilon y.s(x,y)$, as the variable x is bound in the containing ϵ-term. Note, that these restrictions ensure, that the ϵ-rule is not applied at all, if there are no ϵ-terms in the original problem. Of course, the ϵ-rule is also sound without these restrictions.

Theorem 4 (Soundness of Calculus with δ^ϵ- and ϵ-Rules). *Let $\phi \in \mathbf{Fm}^0$ be a closed formula. If there is a closed tableau for $\neg\phi$ using the δ^ϵ and ϵ expansion rules, then ϕ is S-valid.*

Proof. The proof follows the proof for the classical free-variable tableau calculus with the δ^+-rule, see [7]. An S-structure $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ is said to *satisfy* a tableau T, if for all variable assignments $\beta : \mathbf{V} \to \mathcal{D}$ there is a branch on which $\mathcal{S}, \beta \models \phi$ for all formulae ϕ on the branch. We must show, that if $\mathcal{S}$ satisfies T, then $\mathcal{S}$ also satisfies any tableau T' constructed by the application of an expansion rule. Here, only the δ^ϵ- and ϵ-rules are interesting.

If T' is constructed by applying the δ^ϵ-rule for a formula $\exists x.\phi$ on a branch B of T, and $\mathcal{S} = (\mathcal{D}, \mathcal{I}, \mathcal{A})$ satisfies T, let $\beta : \mathbf{V} \to \mathcal{D}$ be a variable assignment, and B_0 a branch, such that all formulae on B_0 are valid under $\mathcal{S}, \beta$. If B_0 and B are not the same, the branch B_0 has not changed, and we are finished. Otherwise, we show that the new formula on B is also valid under $\mathcal{S}, \beta$. We know $\mathcal{S}, \beta \models \exists x.\phi$. From Def. 4 we get $\mathcal{S}, \beta\{x \leftarrow \mathcal{A}(\epsilon x.\phi, \beta)\} \models \phi$, and with $\mathrm{val}(\mathcal{S}, \beta, \epsilon x.\phi) = \mathcal{A}(\epsilon x.\phi, \beta)$ and substitutivity, we have $\mathcal{S}, \beta \models [x/\epsilon x.\phi]\phi$, what we needed to show. Note that there can be no problems with collisions between free variables in $\epsilon x.\phi$ and bound variables in ϕ, as any free variables in ϕ must have been introduced by a γ-rule, and are thus new with respect to any quantified variable. The case for $\neg\forall x.\phi$ is, of course, analogous.

If B is extended using the ϵ-rule, yielding two extended branches, and $\mathcal{S}, \beta$ satisfy every formula on B, there are two cases:

1. $\mathcal{S}, \beta \models \exists x.\phi$. Then, due to Def. 4, we have $\mathcal{S}, \beta\{x \leftarrow \mathcal{A}(\epsilon x.\phi, \beta)\} \models \phi$, and as in the δ^ϵ-case, it follows that $\mathcal{S}, \beta \models [x/\epsilon x.\phi]\phi$. So $\mathcal{S}, \beta$ satisfy all formulae on the right branch.
2. $\mathcal{S}, \beta \not\models \exists x.\phi$. Then we obviously get $\mathcal{S}, \beta \models \forall x.\neg\phi$, and $\mathcal{S}, \beta$ satisfy all formulae on the left branch.

The rest of the proof is identical to the one without ϵ-terms. $\square$

If the formula ϕ to be proved does not contain ϵ-terms, the ϵ-rule can never be applied, so this theorem also proves the soundness of the δ^ϵ-rule, if ϵ-terms are used for $\exists$-quantifier elimination only. Also note that the restrictions of the ϵ-rule were not used in this proof.

Theorem 5 (Completeness of Calculus with δ^ϵ- and ϵ-Rules). *Let $\phi \in \mathbf{Fm}^0$ be a closed S-valid formula. Then there is a closed tableau for $\neg\phi$ using the δ^ϵ and ϵ expansion rules.*

We do not give the proof of this theorem here, as it is rather lengthy and technical. A full proof is given in [6]. Here, we only point out the two main difficulties:

- While the Hintikka-set construction proceeds as usual, the definition of an S-structure satisfying all formulae of the Hintikka-set poses some problems: if we chose the set of all closed terms as carrier set, we would have to apply the ϵ-rule to *all possible* closed ϵ-terms to ensure completeness, contrary to the restrictions of the ϵ-rule. So we need to limit ourselves to all closed terms occurring in atomic formulae of the Hintikka-set. But then, it becomes difficult define the structure in a way that ensures substitutivity for all ϵ-terms and not only for the ones constituting the carrier.
- The restrictions of the ϵ-rule make lifting a trifle more complicated: in the ground version, we restrict the application of the ϵ-rule to closed ϵ-terms occurring in atomic formulae on the current branch. When we lift a ground tableau, these closed ϵ-terms may disappear into a free variable that has not yet been instantiated when the ϵ-rule is applied. In this case, we must show, that there must be a corresponding ϵ-term – possibly containing not-yet-instantiated free variables – somewhere else on the branch. This is the case because the free variable in question will at some time be instantiated by unification, so the instance is necessarily 'somewhere' on the branch from the beginning. Of course, the formal proof is a little involved.

4.2 A Complete Calculus for the Extensional Semantics

We have argued, that the extensional semantics is more intuitive than the substitutive one. Thus, it would be good to have a complete calculus for the extensional semantics too. We now present such a calculus, but it will turn out that it is *not* suited for use in an automated theorem prover.

We obtain a complete calculus for the extensional semantics, if we add another tableau expansion rule to the calculus described in Sec. 4.1, namely

$$\frac{}{\neg\forall z.([x/z]\phi \leftrightarrow [y/z]\phi') \quad | \quad \epsilon x.\phi = \epsilon y.\phi'} \ ,$$

referred to as the *ext* expansion rule.[5] Intuitively the rule says that, whenever we can show that the equivalence of two formulae is a consequence of the current branch, we can identify the values of the corresponding ϵ-terms. Together with any complete set of rules for equality handling, this yields a sound and complete calculus for the extensional semantics, as is shown in [6]. (The completeness proof is much easier as that of Theorem 5, as we do not impose any restrictions on the application of the *ext* or ϵ-rule.)

There is a number of problems with the *ext*-rule:

- We currently do not know – though it seems plausible – whether the rule remains complete if we restrict its application to ϵ-terms already occurring on the branch.
- Even if this were the case, it would have to be applied to any *pair* of occurring ϵ-terms, which would give rise to a quadratic number of rule applications.
- The formula introduced on the left branch is a δ-formula, leading to the introduction of another ϵ-term, which would in turn have to be taken into account for the *ext*-rule. Maybe, it is not necessary for completeness to apply the *ext*-rule to these new ϵ-terms, but that is not yet known.
- Most possible applications of the ϵ-rule would be completely useless for a proof, as two formulae are normally *not* equivalent. Each such unnecessary split would at least double the size of the proof.

Clearly, the *ext*-rule is as dangerous for a machine to apply as a non-atomic cut! Unfortunately, there does not seem to be any other way to cope with extensional semantics.

In the setting of an integrated automated and interactive proof system, we decided to adopt the following view: human users may consider ϵ-terms to have extensional semantics. They are given a complete calculus including the *ext*-rule for interactive work. The automated part of the system uses the calculus described in Sect. 4.1, which is not complete for the extensional semantics. But thanks to the Hierarchy Theorem 2, it is sound: any S-valid formula is also E-valid. And we also provide a precise *semantic* characterization of the incompleteness, namely the automated system can find proofs only for theorems that are not only E-valid, but also S-valid.

[5] This rule was designed for a logic with equality. In a logic without equality, extensionality could be handled with a rule like

$$\frac{\psi(\epsilon x.\phi)}{\neg\forall z.([x/z]\phi \leftrightarrow [y/z]\phi') \quad | \quad \psi(\epsilon y.\phi')} \ .$$

5 Conclusion

The idea of eliminating existential quantifiers by means of ϵ-terms is known for decades. So far, however, this concept is not used (on an object language level) in frameworks for automated deduction. Traditionally, most approaches there deal with Skolem terms instead, e.g. in the context of δ-rules. Compared to simple (i.e. earlier) versions of Skolemization, the ϵ-terms seem to be more complicated. During the last years, on the other hand, the investigation into more efficient δ-rules lead to more sophisticated Skolemization techniques. We interpret this evolution as a movement towards ϵ-like behaviour. Therefore, in this paper we proposed to use ϵ-terms themselves in the context of automated theorem proving, as they have several desirable properties. Compared to Skolem terms, the representation of some information about the 'chosen' element is shifted to the level of the object language. Therefore, the origin and usage of that information is made transparent. For the same reason, object language operations like substitution can be applied to this information. (This exactly is the reason for the exponential speedup discussed in Sect. 2.2.)

Moreover, the usage of ϵ-terms enables us to add a property like extensionality, if desired, by just adding a rule to the calculus. This is the consequence of the semantic hierarchy presented in Sect. 3 and the corresponding rules of Sect. 4. We discussed the suitability of these different variants of an ϵ-calculus for automated proof search. Here, we want to add that substitutivity on the one hand and extensionality on the other hand can be seen as the extremes of a range of ϵ-treatments. Between both, there are other possibilities to exploit special cases of (easily checkable) equivalences. An example for this is the usage of the concept of relevant formulae, used in the δ^{**}-rule of Cantone and Nicolosi [4]. We believe that ϵ-terms provide a framework in which many possible approaches to existential quantifier handling may be expressed.

Acknowledgments

We would like to thank Bernhard Beckert for many fruitful discussions about ϵ-terms and Reiner Hähnle, as well as the referees for their helpful comments on drafts of this paper.

References

1. M. Baaz and C.G. Fermüller. Non-elementary speedups between different versions of tableaux. In 4^{th} *International Workshop, TABLEAUX'95*, LNCS 918, 1995.
2. Bernhard Beckert. *Integrating and Unifying Methods of Tableau-based Theorem Proving*. PhD thesis, Universität Karlsruhe, Fakultät für Informatik, 1998.
3. Bernhard Beckert, Reiner Hähnle, and Peter H. Schmitt. The even more liberalized δ-rule in free variable semantic tableaux. In G. Gottlob, A. Leitsch, and D. Mundici, editors, *Proceedings, 3rd Kurt Gödel Colloquium (KGC), Brno, Czech Republic*, LNCS 713, pages 108–119. Springer, 1993.

4. Domenico Cantone and Marianna Nicolosi Asmundo. A further and effective liberalization of the δ-rule in free variable semantic tableaux. In Ricardo Caferra and Gernot Salzer, editors, *Int. Workshop on First-Order Theorem Proving (FTP'98)*, Technical Report E1852-GS-981, pages 86–96. Technische Universität Wien, Austria, 1998. Electronically available from `http://www.logic.at/ftp98`.
5. Melvin C. Fitting. *First-Order Logic and Automated Theorem Proving*. Springer, 1990.
6. Martin Giese. Integriertes automatisches und interaktives Beweisen: Die Kalkülebene. Diploma Thesis, Fakultät für Informatik, Universität Karlsruhe, June 1998.
7. Reiner Hähnle and Peter H. Schmitt. The liberalized δ-rule in free variable semantic tableaux. *Journal of Automated Reasoning*, 13(2):211–222, 1994.
8. D. Hilbert and P. Bernays. *Grundlagen der Mathematik*, volume 2. Springer Verlag, 1968.
9. A.C. Leisenring. *Mathematical Logic and Hilbert's ϵ-Symbol*. MacDonald, London, 1969.
10. W.P.M. Meyer Viol. *Instantial Logic. An Investigation into Reasoning with Instances*. Number DS–1995–11 in ILLC Dissertation Series. Universiteit van Amsterdam, 1995.
11. Raymond M. Smullyan. *First-Order Logic*, volume 43 of *Ergebnisse der Mathematik und ihrer Grenzgebiete*. Springer-Verlag, New York, 1968.