Translation of Resolution Proofs into Short First-Order Proofs without Choice Axioms

Hans de Nivelle

Max Planck Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken, Germany
nivelle@mpi-sb.mpg.de

Abstract. We present a way of transforming a resolution proof containing Skolemization steps into a natural deduction proof without Skolemization of the same formula. The size of the proof increases only moderately (polynomially). This makes it possible to translate the output of a resolution theorem prover into a purely first-order proof that is moderate in size.

If one wants a resolution based theorem prover to generate explicit proofs, one has to decide what to do with Skolemization. One possibility is to allow Skolemization, (or equivalently the axiom of choice) as a proof principle. In that case, the resolution proof can be translated more or less one-to-one into a natural deduction proof. In [10] it is described how to do this efficiently for the Clausal Normal Form (CNF) transformation. In [6] and [7], a hybrid method was developed. For resolution on the clause level, explicit proofs were generated. For the CNF-transformation, an algorithm was developed inside COQ and proven correct. Using this approach, explicit generation of proofs for the CNF-transformation could be avoided. (Although strictly seen, inside COQ, the term defining the algorithm also defines a proof principle.) A related approach was taken in [14], using the Boyer-Moore theorem prover instead of COQ. Both approaches use the axiom of choice. In [6], the axiom of choice was used for proving the clausification algorithm correct. In [14], a finite domain assumption is used, which implies the axiom of choice.

Another possibility is to completely eliminate the Skolemization steps from the proof. If one is interested in correctness only, the axiom of choice is certainly acceptable, but it is much more elegant to avoid using the axiom of choice in proofs of first-order formulas. Until now, the only way of eliminating applications of Skolemization from a proof, were the methods based on Herbrand's theorem. As a consequence, these methods can cause a hyperexponential increase in proof size, see [20] or [18], see also [4]. In [19], such an algorithm is described in detail. In [13], an improved method is given, which is optimized towards readability of the resulting proof. This method has been implemented in the omega system by Andreas Meijer.

In [1], a general method for eliminating Skolem functions from first-order proofs is given, which results in proofs of polynomial size. The method is based on internalization of a forcing argument. It assumes that there is a theory strong enough to encode finite functions. It is not clear whether this method can be effectively implemented. The problem whether Skolem functions can be efficiently eliminated from every first-order logic proof seems to be open, see the table in [8], Page 9.

In this paper, we give a general method for eliminating Skolem functions from *resolution proofs*, which can be implemented and expected to be efficient. Moreover, it is structure-preserving, by which we mean that it does almost not change the structure of the proof. The main idea is the following: Suppose that f is a Skolem function in the clausal formula $\forall x\ p(x) \vee q(f(x))$. Then f can be replaced by a binary relation F as follows: $\forall x\ \alpha\ F(x, \alpha) \rightarrow p(x) \vee q(\alpha)$. If one makes these replacements in a resolution proof, then the result will be still a valid first-order proof. The surprising fact is that resolution does not make use of the functionality of F, only of the seriality. It will turn out that also proofs containing paramodulation steps can be handled. There is only one restriction on paramodulation, namely that it has to be *simultaneous in the Skolem functions.* Simultaneous in the Skolem functions means that whenever an equality $t_1 \approx t_2$ is applied inside a Skolem term, *all* instances of t_1 that are inside some Skolem term have to be replaced by t_2. The completeness of this restriction follows from the fact that one does not have to paramodulate at all into Skolem terms for completeness. This was proven in [5]

1 Preliminaries

Definition 1. *We assume a fixed set of predicate symbols $\mathcal{P}$ and a fixed set of function symbols $\mathcal{F}$. The sets $\mathcal{P}$ and $\mathcal{F}$ are assumed disjoint. We assume a fixed function* ar, *that attaches to each $f \in \mathcal{F}$ a natural number* $\mathrm{ar}(f) \geq 0$. *In addition,* ar *attaches to each $p \in \mathcal{P}$ a natural number* $\mathrm{ar}(p) \geq 0$. *We assume that for each $n \geq 0$ there are countably infinitely many elements $f \in \mathcal{F}$ with* $\mathrm{ar}(f) = n$.

Similarly, we assume that for each $n \geq 0$, there are countably infinitely many elements $p \in \mathcal{P}$ with $\mathrm{ar}(p) = n$.

We assume that there is no syntactic distinction between variables and constants. We call the elements $c \in \mathcal{F}$, for which $\mathrm{ar}(c) = 0$, either *constants* or *variables* depending on how they are used.

Definition 2. *We recursively define the set of* terms. *If $n \geq 0$, $t_1, \ldots, t_n$ are terms, $f \in \mathcal{F}$ and* $\mathrm{ar}(f) = n$, *then $f(t_1, \ldots, t_n)$ is also a term.*

Next we define the set of atoms. *If $n \geq 0$, $t_1, \ldots, t_n$ are atoms, $p \in \mathcal{P}$ and* $\mathrm{ar}(p) = n$, *then $p(t_1, \ldots, t_n)$ is an atom. If t_1, t_2 are terms, then $t_1 \approx t_2$ is an atom. Formulas are recursively defined as follows:*

- *If A is an atom, then A is also a formula,*
- *$\bot$ and $\top$ are formulas,*
- *if A is a formula, then $\neg A$ is also a formula,*

- *if A, B are formulas, then $A \wedge B$, $A \vee B$, $A \rightarrow B$, $A \leftrightarrow B$ are also formulas,*
- *if $x \in \mathcal{F}$ with $\mathrm{ar}(x) = 0$ and A is a formula, then $\forall x\ P$ and $\exists x\ P$ are also formulas.*

For our purpose, it is convenient to define clauses as a subset of formulas:

Definition 3. *If A is an atom, then the formulas A and $\neg A$ are* literals*. A literal of form A is called* positive*. A literal of form $\neg A$ is called* negative*.*

If $F_1, \ldots, F_n$, are formulas with $n > 0$, then $F_1 \vee \cdots \vee F_n$ simply denotes the disjunction of $F_1, \ldots, F_n$. In case that $n = 0$, $F_1 \vee \cdots \vee F_n$ denotes $\perp$.

A clause *is a formula of form $\forall x_1 \cdots x_k\ \ L_1 \vee \cdots \vee L_n$ in which $L_1, \ldots, L_n$ are literals. We assume that the x_i are distinct. The clause is* empty *if $n = 0$.*

Definition 4. *Let $\mathcal{S} \subseteq \mathcal{P} \cup \mathcal{F}$. For each of the objects defined before (formula, term, atom, literal, clause), we call it an object* over $\mathcal{S}$ *if it contains only predicate and function symbols from $\mathcal{S}$.*

Since we are going to replace function symbols by relations, we need to formally define what a relation is.

Definition 5. *If F is a formula and $x_1, \ldots, x_k \in \mathcal{F}$ have $\mathrm{ar}(x_i) = 0$, then the expression*

$$R = \lambda x_1 \cdots \lambda x_k\ F$$

is a k-ary relation. We also write $\mathrm{ar}(R) = k$.
If $t_1, \ldots, t_n$ are terms, then $R(t_1, \ldots, t_n)$ denotes the formula

$$R[x_1 := t_1]\ [x_2 := t_2] \cdots [x_k := t_k].$$

The notation $[x_i := t_i]$ denotes capture avoiding substitution*.*

The λ-symbol will not occur in the proofs that we construct, because relations will be always instantiated in proofs.
We now define function replacements. In order to define a function replacement, one needs to specify the function symbols that will be replaced. Terms that have a such function symbol on top will be replaced by fresh variables. As a consequence, one also needs to specify a set of fresh variables that will be big enough.

Definition 6. *We write $\mathcal{F}_{\mathrm{Prob}}$ for the set of function symbols occurring in the orginal problem (and its proof), which is a subset of $\mathcal{F}$. We assume a subset $\mathcal{F}_{\mathrm{Repl}}$ of $\mathcal{F}_{\mathrm{Prob}}$, specifying the function symbols that will be replaced. Let $\mathcal{F}_{\mathrm{Def}}$ with $\mathcal{F}_{\mathrm{Prob}} \cap \mathcal{F}_{\mathrm{Def}} = \emptyset$ be the set of variables that will be used as definitions. We use greek letters $\alpha, \beta, \gamma, \delta$ to denote elements of $\mathcal{F}_{\mathrm{Def}}$.*
The function replacement *is a function $[\ ]$ that*

- *assigns to each $f \in \mathcal{F}_{\mathrm{Repl}}$ a relation R_f, s.t. $\mathrm{ar}(R_f) = \mathrm{ar}(f) + 1$.*
- *assigns to each term $f(t_1, \ldots, t_n)$ with $f \in \mathcal{F}_{\mathrm{Repl}}$, a unique element $\alpha \in \mathcal{F}_{\mathrm{Def}}$.*

Definition 7. *Let* $\mathcal{F}_{\text{Repl}}$, $\mathcal{F}_{\text{Def}}$ *and* [ ] *be defined as in Definition 6. The function* [ ] *is extended to terms over* $\mathcal{F}_{\text{Prob}}$ *as follows: The range of extended* [ ] *is the set of terms of terms over* $(\mathcal{F}_{\text{Prob}} \backslash \mathcal{F}_{\text{Repl}}) \cup \mathcal{F}_{\text{Def}}$.

- *For a term* $f(t_1, \ldots, t_n)$ *with* $f \in \mathcal{F}_{\text{Repl}}$, *the replacement* $[f(t_1, \ldots, t_n)]$ *is as defined by Definition 6.*
- *For a term* $f(t_1, \ldots, t_n)$ *with* $f \in \mathcal{F}_{\text{Prob}} \backslash \mathcal{F}_{\text{Repl}}$, *the replacement* $[f(t_1, \ldots, t_n)]$ *is defined as* $f(\ [t_1], \ldots, [t_n]\ )$.

For a quantifier free formula F, *we define* $[F]$ *as the result of replacing each term* t *in* F *by its corresponding* $[t]$.
For a quantifier free formula F, *we define*

- *the set* $\text{Var}(F)$ *as*

$$\{\alpha \in \mathcal{F}_{\text{Def}} \mid \exists t' \in F, \ s.t.\ \alpha = [t']\ \}.$$

These are all the variables of $\mathcal{F}_{\text{Def}}$ *that were involved in defining a subterm of* F *(or* t*).*
- *the* definition set $\text{Def}(A)$ *as the set*

$$\{\ [f](\ [t_1], \ldots, [t_n],\ \alpha) \mid \alpha \in \text{Var}(A), \quad \alpha = [\ f(t_1, \ldots, t_n)\ ]\ \}.$$

For a term t, *the notions* $\text{Var}(t)$ *and* $\text{Def}(t)$ *are defined correspondingly. For a sequence of quantifier free formulas and terms* $U_1, \ldots, U_n$ *(possibly mixed), we define* $\text{Var}(U_1, \ldots, U_n) = \text{Var}(U_1) \cup \cdots \cup \text{Var}(U_n)$, *and* $\text{Def}(U_1, \ldots, U_n) = \text{Var}(U_1) \cup \cdots \cup \text{Var}(U_n)$.

Lemma 1. *For each term* t' *over* $(\mathcal{F}_{\text{Prob}} \backslash \mathcal{F}_{\text{Repl}}) \cup \mathcal{F}_{\text{Def}}$, *there is exactly one term* t *over* $\mathcal{F}_{\text{Prob}}$, *s.t.* $[t] = t'$.

For a variable free formula F, $\text{Var}(F)$ and $\text{Def}(F)$ depend only on the terms in F, and not on the formula structure of F. Therefore it is possible to write $\text{Var}(F, G)$ instead of $\text{Var}(F \wedge G)$ or $\text{Def}(t_1, t_2, F)$ instead of $\text{Def}(t_1 \approx t_2 \vee F)$, etc.

Example 1. Let F be the atomic formula $p(s(f(s(f(0))))\ )$. Assume that $\mathcal{F}_{\text{Repl}} = \{f\}$ and $[f] = R_f$. Further assume that $[f(0)] = \alpha$, $[f(s(f(0)))] = \beta$. Then

$$[s(f(s(f(0))))] = s(\beta), \qquad [f(s(f(0)))] = \beta,$$
$$[s(f(0))] = s(\alpha), \qquad [f(0)] = \alpha,$$
$$[0] = 0.$$

$\text{Var}(F) = \{\alpha, \beta\}$. $\text{Def}(F) = \{R_f(0, \alpha),\ R_f(s(\alpha), \beta)\}$.

We will use the previous definitions to replace a clause

$$C = \forall x_1 \cdots x_k \ \ L_1 \vee \cdots \vee L_n$$

by

$$\forall x_1 \cdots x_k \ \ \forall\ \text{Var}(L_1, \ldots, L_n) \ \bigwedge \text{Def}(L_1, \ldots, L_n) \rightarrow [L_1] \vee \cdots \vee [L_n].$$

We will usually not write the $\bigwedge$-symbol. The replacement [ ] can be chosen in such a way that it replaces Skolem functions by relations.

Example 2. Let $C = \forall x\ p(x, f(x)) \vee q(f(f(x)), x)$ be a clause. Assume that $\mathcal{F}_{\mathrm{Repl}} = \{f\},\ \ [f] = R_f,\ \ [f(x)] = \alpha$, and $[f(f(x))] = \beta$. Then the translation of C equals

$$\forall x\ \forall \alpha\beta\ R_f(x, \alpha) \wedge R_f(\alpha, \beta) \to p(x, \alpha) \vee q(\beta, x).$$

It may appear strange that the value of [ ] depends on the syntactic appearance of a term. For example, one has $[\ f(x)\ ] = \alpha,\ [\ f(y)\ ] = \beta$, while at the same time, the clauses $\forall x\ p(f(x))$ and $\forall y\ p(f(y))$ are α-equivalent.

In fact, one needs to replace the same term by the same variable only within the same clause. One could define a distinct replacement function $[\ ]_C$ for each clause C. However, this would only complicate the presentation of translations in the next section, without introducing more generality.

In practice, if one implements the translation method, it may be impractical to construct a global replacement function, because it becomes too big.

2 Term Replacement

In this section we explain how paramodulation behaves in combination with function replacements. The results in this section contain the essence of the translation method. We define three related concepts, and show that they have related properties. The concepts are *substitutions, generalized substitutions and systems of equations.* A substitution is defined as usual. It assigns terms to variables. In the context of a function replacement, it has to be extended to the variables in $\mathcal{F}_{\mathrm{Def}}$, which is unproblematic.

A generalized substitution is a set of replacement rules of form $t := u$, where t and u are arbitrary terms. When it is applied, *every occurrence* of t has to be replaced by u. Using generalized substitutions, it is possible to define *simultaneous paramodulation.* In [12], it was shown that Skolem functions can be eliminated from resolution proofs all paramodulation steps are simultaneous.

In this paper, we show that it is possible to use a more general form of paramodulation, which we call *non-separating paramodulation.* In non-separating paramodulation, replacement is controlled by *extensions* of *systems of equations.* Roughly speaking, non-separating paramodulation means that it is not allowed to introduce a distinction between two Skolem terms by equality replacement.

Example 3. Consider the equality $0 \approx 1$, and the clause $p(f(0), 0)$. Assume that $\mathcal{F}_{\mathrm{Repl}} = \{f\},\ \ [f] = F$, and that $[f(0)] = \alpha,\ [f(1)] = \beta$. The translation of $p(f(0), 0)$ (as a clause) equals

$$\forall \alpha\ F(0, \alpha) \to p(\alpha, 0).$$

If one paramodulates from the equallity $0 = 1$, one can obtain each of the clauses

$$\begin{array}{ll} p(f(1), 0), & \forall \beta\ F(1, \beta) \to p(\beta, 0), \\ p(f(0), 1), & \forall \alpha\ F(0, \alpha) \to p(\alpha, 1), \\ p(f(1), 1), & \forall \beta\ F(1, \beta) \to p(\beta, 1). \end{array}$$

Example 4. Now consider the equality $f(0) \approx f(1)$, and the same clause $p(f(0), 1)$. Let $\mathcal{F}_{\text{Repl}}$ and [ ] be as in the previous example. Then one has:

$$\begin{array}{ll} f(0) \approx f(1) & \forall\alpha\ \beta\ F(0,\alpha) \to F(1,\beta) \to \alpha \approx \beta \\ p(f(0), f(0)) & \forall\alpha\ F(0,\alpha) \to p(\alpha,\alpha) \\ p(f(0), f(1)) & \forall\alpha\ \beta\ F(0,\alpha) \to F(1,\beta) \to p(\alpha,\beta) \end{array}$$

and the formulas in the 3d row are derivable other formulas.

The examples show the principle of how paramodulation steps can be reconstructed after translation by a function replacement. If some term t_1 occurs inside some literal R, then $[t_1]$ occurs either in $[R]$ or in $\text{Def}(R)$, and the replacement can be made there.

Example 5. Consider the equality $0 \approx 1$, and the clause $p(f(0), f(0))$. Let [ ] and $\mathcal{F}_{\text{Repl}}$ be defined as in the previous example. From $\forall\alpha\ F(0,\alpha) \to p(\alpha,\alpha)$ and $0 \approx 1$, one can prove $\forall\beta\ F(1,\beta) \to p(\beta,\beta)$, but not $\forall\alpha\ \beta\ \ F(0,\alpha) \to F(1,\beta) \to p(\alpha,\beta)$.

The last example shows the major problem when translating arbitrary paramodulation steps. If one wants to replace t_1 by t_2 inside some atom R, and $[t_1]$ occurs in $\text{Def}(R)$, then all subterms that depend on the occurrence $[t_1]$ will be automatically modified. Because of this reason, only simultaneous paramodulation was considered in [12]. In this paper, we show that a weaker restriction of paramodulation will work as well.

We now define both substitutions and generalized substitutions, and how they are translated by a function replacement [ ].

Definition 8. *A* substitution *is a set of form* $\Theta = \{x_1 := t_1, \ldots, x_k := t_k\}$, *s.t.* $(x_{i_1} = x_{i_2}) \Rightarrow (t_{i_1} = t_{i_2})$. *Each* x_i *is a variable, and each* t_i *is a term.*
The application of Θ on a term t, *notation* $t \cdot \Theta$, *is recursively defined as follows (in the standard way):*

- *if* t *equals one of the* x_i, *then* $t \cdot \Sigma = t_i$.
- *Otherwise, write* $t = f(w_1, \ldots, w_n)$. *The application* $f(w_1, \ldots, w_n) \cdot \Sigma$ *equals* $f(w_1 \cdot \Sigma, \ldots, w_n \cdot \Sigma)$.

The application of Θ *on a quantifier free formula* F *is defined term wise.*

Definition 9. *A* generalized substitution *is a set of form* $\Sigma = \{t_1 := u_1, \ldots, t_k := u_k\}$, *s.t. there exist no* i_1, i_2 *with* $1 \leq i_1, i_2 \leq k$, *and* t_{i_1} *is a subterm of* t_{i_2}. *The application of* Σ *on a term* t, *notation* $t \cdot \Sigma$, *is recursively defined as follows:*

- *If* t *equals one of the* t_i, *then* $t \cdot \Sigma = u_i$.
- *Otherwise, write* $t = f(w_1, \ldots, w_n)$. *The application* $f(w_1, \ldots, w_n) \cdot \Sigma$ *equals* $f(w_1 \cdot \Sigma, \ldots, w_n \cdot \Sigma)$.

The application of Σ *on a quantifier free formula* F *is defined term wise.*

Substitutions and generalized substitutions are closely related. One could say that substitutions are 'a subclass' of generalized substitutions. We now define how a function replacement [ ] translates a generalized substitution. By 'inheritance', the translation also applies to simple substitutions.

Definition 10. *Let $\Sigma = \{t_1 := u_1, \ldots, t_k := u_k\}$ be a generalized substitution on terms over $\mathcal{F}_{\mathrm{Prob}}$. Let [ ] be a function replacement, replacing functions from $\mathcal{F}_{\mathrm{Repl}} \subseteq \mathcal{F}_{\mathrm{Prob}}$ and introducing variables from $\mathcal{F}_{\mathrm{Def}}$.*

We define the replacement of Σ, *for which we write $[\Sigma]$, as the union of a substitution and a generalized substitution. The first one, $[\Sigma]_{\mathrm{Prob}}$ contains the straightforward translation of Σ by [ ]. The second one, $[\Sigma]_{\mathrm{Def}}$, defines the translation of the application operator on $\mathcal{F}_{\mathrm{Def}}$.*

$$[\Sigma]_{\mathrm{Prob}} = \{\ [t_1] := [u_1], \ldots, [t_k] := [u_k]\ \}$$

$$[\Sigma]_{\mathrm{Def}} = \{\ \alpha := [t \cdot \Sigma] \mid \alpha \in \mathcal{F}_{\mathrm{Def}},\quad \alpha = [t] \text{ and } t \neq t \cdot \Sigma\ \}.$$

$$[\Sigma] = [\Sigma]_{\mathrm{Prob}} \cup [\Sigma]_{\mathrm{Def}}.$$

Note that the notation $[\Sigma]_{\mathrm{Prob}}$ is slightly misleading, because the $[t_i]$ and $[u_j]$ can contain variables from $\mathcal{F}_{\mathrm{Def}}$ as well. It is easily checked that $[\Sigma]$ is always a well-formed, generalized substitution.

Theorem 1. *Let $\Sigma = \{t_1 := u_1, \ldots, t_k := u_k\}$ be a generalized substitution on terms over $\mathcal{F}_{\mathrm{Prob}}$. Let [ ] be a function replacement, replacing functions from $\mathcal{F}_{\mathrm{Repl}} \subseteq \mathcal{F}_{\mathrm{Prob}}$, and introducing variables from $\mathcal{F}_{\mathrm{Def}}$. For every term term t over $\mathcal{F}_{\mathrm{Prob}}$,*

$$[t \cdot \Sigma] = [t] \cdot [\Sigma].$$

Proof. We use induction on the term structure of t.

- If t equals one of the t_i, then $[t_i \cdot \Sigma] = [u_i]$, by construction of $[\Sigma] \supseteq [\Sigma]_{\mathrm{Prob}}$.
- If t does not equal any of the t_i, and $[t] \in \mathcal{F}_{\mathrm{Def}}$, then $[t] \cdot [\Sigma] = [t \cdot \Sigma]$, by construction of $[\Sigma] \supseteq [\Sigma]_{\mathrm{Def}}$.
- If t does not equal any of the t_i, and $[t] \notin \mathcal{F}_{\mathrm{Def}}$, then write $t = g(w_1, \ldots, w_n)$. We have $[g(w_1, \ldots, w_n) \cdot \Sigma] = [g(w_1 \cdot \Sigma, \ldots, w_n \cdot \Sigma)] = g([w_1 \cdot \Sigma], \ldots, [w_n \cdot \Sigma])$. By induction, this equals $g([w_1] \cdot [\Sigma], \ldots, [w_n] \cdot [\Sigma])$. But this is equal to $[g(w_1, \ldots, w_n)] \cdot [\Sigma]$, because $[t] \neq [t_i]$ and $[t] \notin \mathcal{F}_{\mathrm{Def}}$.

Theorem 2. *Let $\Sigma = \{t_1 := u_1, \ldots, t_k := u_k\}$ be a generalized substitution on terms over $\mathcal{F}_{\mathrm{Prob}}$. Let [ ] be a function replacement, replacing functions from $\mathcal{F}_{\mathrm{Repl}} \subseteq \mathcal{F}_{\mathrm{Prob}}$, and introducing variables from $\mathcal{F}_{\mathrm{Def}}$.*
For every term t over $\mathcal{F}_{\mathrm{Prob}}$,

$$[t_1] \approx [u_1], \ldots, [t_k] \approx [u_k] \vdash ([t] \cdot [\Sigma]) \approx ([t] \cdot [\Sigma]_{\mathrm{Def}}).$$

Proof. The missing replacements can be made up by equality replacement.

In the rest of this paper, we will only use substitutions, not generalized substitutions. Theorem 2 will not be used.

Example 6. Consider the equality $0 \approx 1$, and the clause $p(f(0,0), f(0,0))$. Assume that $\mathcal{F}_{\text{Repl}} = \{f\}$, $[f] = F$, and that

$$[f(0,0)] = \alpha, \quad [f(0,1)] = \beta, \quad [f(1,0)] = \gamma, \quad [f(1,1)] = \delta.$$

Using [ ], the clause $p(f(0,0), f(0,0))$ translates into

$$\forall \alpha \; F(0,0,\alpha) \to p(\alpha, \alpha).$$

Using paramodulation from $0 \approx 1$, the following 3 clauses can be obtained:

$$\forall \beta \; F(0,1,\beta) \to p(\beta, \beta),$$
$$\forall \gamma \; F(1,0,\gamma) \to p(\gamma, \gamma),$$
$$\forall \delta \; F(1,1,\delta) \to p(\delta, \delta).$$

The following clauses are examples of clauses that cannot be obtained:

$$\forall \alpha\beta \; F(0,0,\alpha) \to F(0,1,\beta) \to p(\alpha, \beta),$$
$$\forall \beta\gamma \; F(0,1,\beta) \to F(1,0,\gamma) \to p(\beta, \gamma),$$

Example 6 shows that one does not always have to replace all occurrences. Nevertheless, one also does not have a full freedom when deciding which occurrences are to be replaced. If one wants to paramodulate from an equation $t_1 \approx t_2$ into a literal A, then the possibilities are determined by the occurrences of $[t_1]$ in $[A]$. In the clause of Example 6, the first and second occurrence of 0 are represented by distinct arguments of F. However the (first and third), and the (second and fourth) occurrence are represented by the same argument of F. Therefore these cannot be separated.

Whenever some term, constructed by a function symbol in $\mathcal{F}_{\text{Repl}}$, has more than one occurrence, all occurrences are represented by the same variable in the [ ]-translation. Therefore, paramodulation must be carried out in such a way that it does not introduce a distinction between identical terms with a symbol from $\mathcal{F}_{\text{Repl}}$ on top. In the previous example, $f(0,0)$ was such a term. We call the resulting restriction of paramodulation *non-separating*.

We now define the notion of *system of equations*. Only systems with $k = 1$ will be used in this paper, but the results that we prove in this section also hold for $k > 1$.

Definition 11. *A* system of equations $\mathcal{E}$ *is a set of form* $\mathcal{E} = \{u_1 \approx t_1, \ldots, u_k \approx t_k\}$, *where* $u_1, t_1, \ldots, u_k, t_k$ *are terms.*

Replacement of equals is controlled by *extensions*. An extension determines how replacements are made inside identical $\mathcal{F}_{\text{Repl}}$-terms.

Definition 12. *Let* $\mathcal{E} = \{u_1 \approx t_1, \ldots, u_k \approx t_k\}$ *be a system of equations with terms over* $\mathcal{F}_{\text{Prob}}$. *Let* t *and* u *be two terms over* $\mathcal{F}_{\text{Prob}}$. *We write* $\mathcal{E}(t, u)$ *if* u *can be obtained from* t *by finitely often replacing a* t_i *by its* u_i *(or a* u_i *by its* t_i*), at arbitrary positions, but never in the scope of a function* $f \in \mathcal{F}_{\text{Repl}}$.
An extension Σ *of* $\mathcal{E}$ *is a function from the set of terms over* $\mathcal{F}_{\text{Prob}}$ *to itself. For every term* $f(w_1, \ldots, w_n)$ *over* $\mathcal{F}_{\text{Prob}}$, *the following recursive condition must hold:*

– *If* $f \in \mathcal{F}_{\mathrm{Repl}}$, *then*

$$f(w_1, \ldots, w_n) \cdot \Sigma \textit{ has form } f(v_1, \ldots, v_n),$$

and for each i *with* $1 \leq i \leq n$, *it must be the case that*

$$\mathcal{E}(\ v_i,\ \ w_i \cdot \Sigma\ ).$$

– *If* $f \notin \mathcal{F}_{\mathrm{Repl}}$, *then*

$$f(w_1, \ldots, w_n) \cdot \Sigma = f(w_1 \cdot \Sigma, \ldots, w_n \cdot \Sigma).$$

The application of Σ *on a quantifier free formula* F *is obtained by applying* Σ *on each top level term in* F.

We write $t \cdot \Sigma$ instead of $\Sigma(t)$, because of the close relation with the extension of a generalized substitution. The $\mathcal{E}$-relation allows arbitrary replacement of equals by equals, but not in the scope of a function symbol from $\mathcal{F}_{\mathrm{Repl}}$. Replacements inside the scope of a function symbol from $\mathcal{F}_{\mathrm{Repl}}$ are controlled by the extension, which ensures that the same term is always rewritten in the same way. The non-separating paramodulation rule is defined in Definition 19.

Example 7. In the first paramodulant of Example 6, $\mathcal{E} = \{0 \approx 1\}$, and $f(0,0) \cdot \Sigma = f(0,1)$. For the atom $p(f(0,0), f(0,0))$, the only atom A with $\mathcal{E}(\ p(f(0,0), f(0,0)), A)$ equals $p(f(0,1), f(0,1))$. For atom $q(f(0,0), 0, f(0,0))$, there would be two possibilities, $q(f(0,1), 0, f(0,1))$ and $q(f(0,1), 1, f(0,1))$.

Definition 13. *Let* $\mathcal{E} = \{t_1 \approx u_1, \ldots, t_k \approx u_k\}$ *be a system of equations with terms over* $\mathcal{F}_{\mathrm{Prob}}$. *The* replacement *of* $\mathcal{E}$, *written as* $[\mathcal{E}]$, *is defined as the system of equations*

$$\{\ [t_1] \approx [u_1], \ldots, [t_k] \approx [u_k]\ \}.$$

Let Σ *be an extension of* $\mathcal{E}$. *The* replacement $[\Sigma]$ *of* Σ *is defined as the substitution*

$$[\Sigma] = \{\ \alpha := [t \cdot \Sigma] \mid \alpha \in \mathcal{F}_{\mathrm{Def}},\ \textit{and } \alpha = [t]\ \}.$$

Theorem 3. *For every term* t *over* $\mathcal{F}_{\mathrm{Prob}}$,

$$[t \cdot \Sigma] = [t] \cdot [\Sigma].$$

For every quantifier free formula F *with terms over* $\mathcal{F}_{\mathrm{Prob}}$,

$$[F \cdot \Sigma] = [F] \cdot [\Sigma].$$

Theorem 4. *For each pair* w_1, w_2 *of terms over* $\mathcal{F}_{\mathrm{Prob}}$,

$$\mathcal{E}(w_1, w_2) \textit{ implies } [\mathcal{E}] \vdash [w_1] \approx [w_2].$$

Proof. It is enough to show the lemma under the assumption that w_2 can be obtained from w_1 by a single replacement. Suppose that there is an equation $(t \approx u) \in \mathcal{E}$, s.t. there is a position π in w_1 and w_2, s.t. w_1 and w_2 differ only at position π, w_1 contains t on π, and w_2 contains u on π., Then $([t] \approx [u]) \in [\mathcal{E}]$, $[w_1]$ and $[w_2]$ differ only at position π, $[w_1]$ contains $[t]$ on π and $[w_2]$ contains $[u]$ on π.

Theorem 5. *Let $\mathcal{E} = \{t_1 \approx u_1, \ldots, t_k \approx u_k\}$ be a system of equations with terms over $\mathcal{F}_{\text{Prob}}$. Let Σ be an extension of $\mathcal{E}$. Let z_1 be a term over $\mathcal{F}_{\text{Prob}}$, which is constructed by a function symbol $f \in \mathcal{F}_{\text{Repl}}$. Let z_2 be some term over $\mathcal{F}_{\text{Prob}}$ that contains z_1.*
Then either $z_1 \cdot \Sigma$ is a subterm of $z_2 \cdot \Sigma$, or $z_1 \cdot \Sigma$ is a subterm of one of the terms $t_1, u_1, \ldots, t_k, u_k$.

Proof. Suppose that $z_1 \cdot \Sigma$ is not a subterm of $z_2 \cdot \Sigma$. Let z' be a smallest subterm of z_2, s.t.

- z_1 is a strict subterm of z' and z' is a subterm of z_2,
- z' has form $f(w_1, \ldots, w_n)$ with $f \in \mathcal{F}_{\text{Repl}}$.
- $z_1 \cdot \Sigma$ is not contained in $z' \cdot \Sigma$.

We show that such z' exists. First, let z'' be a smallest subterm of z_2 which contains z_1 and for which $z_1 \cdot \Sigma$ is not a subterm of $z'' \cdot \Sigma$. Then, if all subterms between z_1 and z'' would have a function symbol $f \not\in \mathcal{F}_{\text{Repl}}$ on top, then $z_1 \cdot \Sigma$ would be a subterm of $z'' \cdot \Sigma$, because $f(w_1, \ldots, w_n) \cdot \Sigma = f(w_1 \cdot \Sigma, \ldots, w_n \cdot \Sigma)$ in case that $f \not\in \mathcal{F}_{\text{Repl}}$,

The application $f(w_1, \ldots, w_n) \cdot \Sigma$ has form $f(v_1, \ldots, v_n)$. Term z_1 is a subterm of one of the w_i. By definition of extension, $\mathcal{E}(\ w_i \cdot \Sigma,\ v_i\ )$. By minimality of z', it must be the case that $z_1 \cdot \Sigma$ is a subterm of $w_i \cdot \Sigma$. From the construction of z', it follows that $z_1 \cdot \Sigma$ is not a subterm of v_i. Because $\mathcal{E}(w_i \cdot \Sigma,\ v_i)$, it is possible to rewrite $w_i \cdot \Sigma$ into v_i, using the equalities in $\mathcal{E}$. In the rewrite sequence, there is a last term that still contains $z_1 \cdot \Sigma$. Because $z_i \cdot \Sigma$ is constructed by a term in $\mathcal{F}_{\text{Repl}}$, rewriting inside $z_1 \cdot \Sigma$ is not allowed. Therefore the equality $t_j \approx u_j$ (with $1 \leq j \leq k$) that removes $z_1 \cdot \Sigma$ must contain $z_1 \cdot \Sigma$.

3 Translation of Resolution on the Clause Level

In this section we will show the following: Let [ ] be some function replacement replacing functions from $\mathcal{F}_{\text{Repl}}$ and introducing variables from $\mathcal{F}_{\text{Def}}$. Let S be some set of clauses. If S has a resolution refutation in which all paramodulation steps are non-separating, and $[S]$ is obtained from S by replacing each clause $\forall x_1 \cdots x_k\ R$ by its translation $\forall x_1 \cdots x_k\ \forall\ \text{Var}(R)\ \text{Def}(R) \rightarrow [R]$, then $[S]$ has a natural deduction refutation with a size bounded by a polynomial in the size of the refutation of S, on the condition that for each function $f \in \mathcal{F}_{\text{Repl}}$, its translation R_f is serial, which means the following:

Definition 14. *Let R be an $(n+1)$-ary relation. The* seriality axiom *for R is the formula $\forall x_1 \cdots x_n\ \exists y\ R(x_1, \ldots, x_n, y)$.*

The refutation of $[S]$ can be obtained by step-by-step translation of the proof steps. We will sum up the standard resolution rules, as they can be found for example in [15], and show that for each rule the translation of the conclusion is provable from the translations of the premisses. We will not explicitly consider the polynomial complexity bound, because it will be evident from the proof constructions that they have small polynomial size.

In order for the translation to work, paramodulation needs to be *non-separating*, which intuitively means that equality replacement cannot introduce a distinction between two Skolem terms in a clause.
In resolution, instantiation is controlled by unification of terms or literals that need to be equal before the rule can be applied. The use of unification is important for efficiency, but not important for soundness of the rules. Therefore we can define a separate instantiation rule, and assume that the other rules do not instantiate. This simplifies the presentation. We define instantiation. The definition is slightly more complicated than usual, because we do not have implicit quantification.

Definition 15. *A* generalization Λ *is a set of form* $\{x_1, \ldots, x_m\}$, *s.t. each* x_i *is a variable. If* F *is a formula, we write* $F \cdot \Lambda$ *for the application of* Λ *on* F*. The result equals* $\forall x_1 \cdots x_m \; F$.

Definition 16. *Let* $C = \forall x_1 \cdots x_k \; R$ *and* $D = \forall y_1 \cdots y_m \; S$ *be clauses. We call* D an instance *of* C *if there exists a substitution* Θ, *which assigns only to variables from* $x_1, \ldots, x_k$, *s.t.* $R \cdot \Theta = S$, *and for the generalization* $\{y_1, \ldots, y_m\}$, *none of the variables* $y_1, \ldots, y_m$ *is free in* $\forall x_1 \cdots x_k \; R$.

It is easily checked that $C \models D$, if D is an instance of C. We also treat permutation separately:

Definition 17. *Clauses* $\forall x_1 \cdots x_k \;\; L_1 \vee \cdots \vee L_m$ *and* $\forall x_1 \cdots x_k \;\; M_1 \vee \cdots \vee M_n$ *are permutations of each other if*

$$\{L_1, \ldots, L_m\} = \{M_1, \ldots, M_n\}.$$

Definition 18. *We define the unary rules:*

equality swapping

$$\frac{\forall x_1 \cdots x_k \;\; t_1 \approx t_2 \vee R}{\forall x_1 \cdots x_k \;\; t_2 \approx t_1 \vee R} \qquad\qquad \frac{\forall x_1 \cdots x_k \;\; t_1 \not\approx t_2 \vee R}{\forall x_1 \cdots x_k \;\; t_2 \not\approx t_1 \vee R}$$

equality reflexivity

$$\frac{\forall x_1 \cdots x_k \;\; t \not\approx t \vee R}{\forall x_1 \cdots x_k \;\; R}$$

equality factoring

$$\frac{\forall x_1 \cdots x_k \;\; t_1 \approx t_2 \vee t_1 \approx t_3 \vee R}{\forall x_1 \cdots x_k \;\; t_1 \approx t_2 \vee t_2 \not\approx t_3 \vee R}$$

Definition 19. *We define the binary rules:*

resolution

$$\frac{\forall x_1 \cdots x_k \;\; A \vee R_1 \qquad \forall x_1 \cdots x_k \;\; \neg A \vee R_2}{\forall x_1 \cdots x_k \;\; R_1 \vee R_2}$$

non-separating paramodulation

Let $\mathcal{E} = \{t_1 \approx t_2\}$. Let Σ be an extension of $\mathcal{E}$. Assume that $\mathcal{E}(\ R_2',\ R_2 \cdot \Sigma\ )$, Then

$$\frac{\forall x_1 \cdots x_k \;\; t_1 \approx t_2 \vee R_1 \qquad \forall x_1 \cdots x_k \;\; R_2}{\forall x_1 \cdots x_k \;\; R_1 \vee R_2'}$$

The intuitive meaning of the non-separating paramodulation rule is as follows: If there there are two occurrences of the same term $f(w_1, \ldots, w_n)$ with $f \in \mathcal{F}_{\text{Repl}}$, which contain t_1 on some position, then t_1 has to be replaced by t_2 either on both positions or on neither of them.

Example 8. Assume that $\mathcal{F}_{\text{Repl}} = \{f, g\}$. Using equality $0 \approx 1$, we have

$p(0,0,0)$	$\Rightarrow$	$p(0,1,1)$	possible,
$q(s(0,0), s(0,0))$	$\Rightarrow$	$q(s(0,1), s(1,0))$	possible because $s \notin \mathcal{F}_{\text{Repl}}$,
$q(f(0,0), f(0,0))$	$\Rightarrow$	$q(f(1,1), f(0,0))$	not possible,
$p(f(0,0), f(0,0), 0)$	$\Rightarrow$	$p(f(0,1), f(0,1), 0)$	possible,
$p(f(0,0), f(0,0), 0)$	$\Rightarrow$	$p(f(1,0), f(1,0), 1)$	possible,
$q(f(0,0), g(0,0))$	$\Rightarrow$	$q(f(0,1), g(1,0))$	possible.

In case one does not paramodulate into Skolem terms, which is known to be complete, and often implemented for efficiency reasons, all paramodulation steps will be automatically non-separating. We provide the translations for the derivation rules:

3.1 Instantiation

Assume that the clause $\forall y_1 \cdots y_m \;\; S$ is an instance of the clause $\forall x_1 \cdots x_k \;\; R$ through substitution Θ and generalization Λ. We need to construct a proof that

$$\forall x_1 \cdots x_k \; \forall \, \text{Var}(R) \;\; \text{Def}(R) \rightarrow [R]$$

implies

$$\forall y_1 \cdots y_m \; \forall \, \text{Var}(S) \;\; \text{Def}(S) \rightarrow [S].$$

Write $\Theta = \{x_1 := t_1, \ldots, x_k := t_k\}$. We have $R \cdot \Theta = S$. The generalization Λ equals $\{y_1, \ldots, y_m\}$, and none of the y_j is free in $\forall x_1 \cdots x_k \;\; R$.

Let $[\Theta]$ be constructed from Θ as in Definition 10. It is easily checked that $(x := t) \in [\Theta]$ implies $x \in \mathcal{F}_{\text{Def}}$ or x is among the $x_1, \ldots, x_k$. As a consequence $[\Theta]$ is a substitution and it is possible to construct the proof given in Figure 1. We justify the proof steps:

S1 Because $y_1, \ldots, y_m$ are not free in $\forall x_1 \cdots x_k \;\; R$, they are also not free in $\forall x_1 \cdots x_k \; \forall \, \text{Var}(R) \;\; \text{Def}(R) \rightarrow [R]$. Therefore, the $y_1, \ldots, y_m$ are fresh.

S2 If a variable $\alpha \in \text{Var}(S)$ occurs in $[R]$, then it also occurs in $\text{Var}(R)$. Hence it is still fresh.

S3 An assumption.

S4 $[\Theta]$ is a well-formed substitution.

S5 It is easily seen that $(\text{Def}(R) \rightarrow [R]) \cdot [\Theta] = (\text{Def}(R) \cdot [\Theta]) \rightarrow ([R] \cdot [\Theta])$, but we also need to check that all atoms in $\text{Def}(R) \cdot [\Theta]$ are provable. Let $A \in \text{Def}(R)$. Then A has form $R_f(\ [w_1], \ldots, [w_n], [f(w_1, \ldots, w_n)]\ )$, where $f \in \mathcal{F}_{\text{Repl}}$ and $f(w_1, \ldots, w_n)$ occurs in R. Because f is not in the domain of Θ, $f(w_1, \ldots, w_n) \cdot \Theta$ occurs in S and equals $f(w_1 \cdot \Theta, \ldots, w_n \cdot \Theta)$. As a consequence, we have the atom $R_f([w_1 \cdot \Theta], \ldots, [w_n \cdot \Theta], [f(w_1, \ldots, w_n) \cdot \Theta]) \in \text{Def}(S)$. From Theorem 1, it follows that this equals

$$R_f(\ [w_1] \cdot [\Theta], \ldots, [w_n] \cdot [\Theta],\ [f(w_1, \ldots, w_n)] \cdot [\Theta]\ ),$$

which in turn equals

$$R_f(\ [w_1], \ldots, [w_n],\ [f(w_1, \ldots, w_n)]\ ) \cdot [\Theta] = A \cdot [\Theta].$$

S6 By Theorem 1, $[R] \cdot [\Theta] = [R \cdot \Theta] = [S]$.

Fig. 1. Natural Deduction Proof for the Instantiation Rule

$\forall x_1 \cdots x_k\ \forall\ \text{Var}(R)\ \text{Def}(R) \rightarrow [R]$

Fresh $y_1 \cdots y_m$	S1
Fresh $\text{Var}(S)$	S2
$\text{Def}(S)$	S3
$(\ \text{Def}(R) \rightarrow [R]\ ) \cdot [\Theta]$	S4
$[R] \cdot [\Theta]$	S5
$[S]$	S6

$\forall y_1 \cdots y_m\ \forall\ \text{Var}(S)\ \text{Def}(S) \rightarrow [S]$

3.2 Equality Reflexivity

Assume that the clause $\forall x_1 \cdots x_k\ R$ is obtained from $\forall x_1 \cdots x_k\ t \not\approx t \vee R$ by equality reflexivity. We need to construct a proof of the fact that

$$\forall x_1 \cdots x_k\ \forall\ \text{Var}(t, R)\ \text{Def}(t, R) \rightarrow\ [t \not\approx t \vee R]$$

implies

$$\forall x_1 \cdots x_k\ \forall\ \text{Var}(R)\ \text{Def}(R) \rightarrow [R].$$

There is no difficulty in showing that $[t] \not\approx [t] \vee [R]$ implies $[R]$. The difficulty of the proof is the fact that there may be variables in $\mathrm{Var}(t, R)$, with corresponding definitions in $\mathrm{Def}(t, R)$, that do not occur in $\mathrm{Var}(R)$ (and $\mathrm{Def}(R)$) For these variables, proper instantiations need to be found. In order to find these, the seriality axioms are needed.
Write

$$\mathrm{Var}(t, R)\backslash\mathrm{Var}(R) = \{\alpha_1, \ldots, \alpha_n\}, \text{ with } n \geq 0.$$

Assume that the α_i are ordered in such a way that if $\alpha_i = [s_1], \quad \alpha_j = [s_2]$, and s_1 is a subterm of s_2, then $i \leq j$. Write $A_1(\overline{w}_1, \alpha_1), A_2(\overline{w}_2, \alpha_2), \ldots, A_n(\overline{w}_n, \alpha_n)$ for $\mathrm{Def}(t, R)\backslash\mathrm{Def}(R)$. Due to the way the $\alpha_1, \ldots, \alpha_n$ are ordered, α_j does not occur in $\overline{w}_i$, if $i \leq j$. Using this, we can construct the proof given in Figure 2, in which the $\alpha_1, \ldots, \alpha_n$ are 'resolved away' with the seriality axioms.

3.3 Resolution, Equality Swapping, Equality Factoring, Permutation

For the other rules, with the exception of paramodulation, it fairly easy to show that they can be reconstructed.

In the resolution rule, it is possible that a term with an $f \in \mathcal{F}_{\mathrm{Repl}}$ as top symbol occurs in one of the premisses, but not in the result. In that case, the definitions for the terms that do not occur in the result need to be resolved away, in the same way as with the equality reflexivity rule. One can either do this directly, or alternatively reformulate the resolution rule as follows:

resolution 2

$$\frac{\forall x_1 \cdots x_k \;\; A \vee R_1 \qquad \forall x_1 \cdots x_k \;\; \neg A \vee R_2}{\forall x_1 \cdots x_k \;\; u_1 \not\approx u_1 \vee \cdots \vee u_n \not\approx u_n \vee R_1 \vee R_2},$$

Here $u_1, \ldots, u_n$ are the subterms that occur in A but not in $R_1 \vee R_2$. After this, $\forall x_1 \cdots x_k \;\; R_1 \vee R_2$ can be obtained through n applications of equality reflexivity.

3.4 Non-Separating Paramodulation

The non-separating paramodulation rule is the rule that is the most complicated to translate:

non-separating paramodulation

$$\frac{\forall x_1 \cdots x_k \;\; t_1 \approx t_2 \vee R_1 \qquad \forall x_1 \cdots x_k \; R_2}{\forall x_1 \cdots x_k \;\; R_1 \vee R_2'}$$

on the condition that $\mathcal{E}(\; R_2',\; R_2 \cdot \Sigma \;)$, with $\mathcal{E} = \{t_1 \approx t_2\}$ and Σ an extension of $\mathcal{E}$.

As is the case with the resolution rule, there can be terms occurring in one of the premisses that do not occur in the conclusion. One can proceed in the same way as with the resolution, by keeping the removed terms in negated equations

Fig. 2. Proof for Equality Reflexivity

$\forall x_1 \cdots x_k\ \forall\, \mathrm{Var}(t,R)\ \mathrm{Def}(t,R) \rightarrow [t \not\approx t \vee R]$ (assumption)

$\forall x_1 \cdots x_k$
$\quad \forall \alpha_1\ A_1(\overline{w}_1, \alpha_1)\ \ \forall \alpha_2\ A_2(\overline{w}_2, \alpha_2)\ \ \cdots\ \ \forall \alpha_n\ A_n(\overline{w}_n, \alpha_n)$
$\qquad \forall\, \mathrm{Var}(R)\ \mathrm{Def}(R) \rightarrow [t \not\approx t \vee R]$ (rearranging quantifiers)

> Fresh $x_1 \cdots x_k$
> Fresh $\mathrm{Var}(R)$
> $\mathrm{Def}(R)$
>
> ---
>
> $\forall \alpha_1\ A_1(\overline{w}_1, \alpha_1)\ \ \forall \alpha_2\ A_2(\overline{w}_2, \alpha_2)\ \ \cdots\ \ \forall \alpha_n\ A_n(\overline{w}_n, \alpha_n)$
> $\qquad \forall\, \mathrm{Var}(R)\ \mathrm{Def}(R) \rightarrow [t \not\approx t \vee R]$ (instantiation)
>
> $\exists \alpha_1\ A_1(\overline{w}_1, \alpha_1)$ (instantiation of seriality axiom for A_1)
>
> > $A_1(\overline{w}_1, \alpha_1)$
> >
> > ---
> >
> > $\forall \alpha_2\ A_2(\overline{w}_2, \alpha_2)\ \ \cdots\ \ \forall \alpha_n\ A_n(\overline{w}_n, \alpha_n)$
> > $\qquad \forall\, \mathrm{Var}(R)\ \mathrm{Def}(R) \rightarrow [t \not\approx t \vee R]$ (instantiation)
> >
> > $\exists \alpha_2\ A_2(\overline{w}_2, \alpha_2)$ (instantiation of seriality axiom for A_2)
> >
> > > $A_2(\overline{w}_2, \alpha_2)$
> > >
> > > ---
> > >
> > > . . . .
> > >
> > > > $\exists \alpha_n\ A_n(\overline{w}_n, \alpha_n)$ (instantiation of seriality axiom for A_n)
> > > >
> > > > > $A_n(\overline{w}_n, \alpha_n)$
> > > > >
> > > > > ---
> > > > >
> > > > > $\forall\, \mathrm{Var}(R)\ \mathrm{Def}(R) \rightarrow [t \not\approx t \vee R]$
> > > > > $[t \not\approx t \vee R]$
> > > > > $[R]$
> > > >
> > > > $[R]$ ($\exists$-elimination)
> > >
> > > . . . .
> > >
> > > $[R]$ ($\exists$-elimination)
> >
> > $[R]$ ($\exists$-elimination)
>
> $[R]$ ($\exists$-elimination)

$\forall x_1 \cdots x_k\ \ \forall\, \mathrm{Var}(R)\ \ \mathrm{Def}(R) \rightarrow [R]$ ($\forall$-introduction, $\rightarrow$-introduction) in the conclusion. However, there is no need to keep the negative equations since their removal is trivial. It is sufficient to keep the definitions of the terms that disappeared. The result is the following rule:

$$\forall x_1 \cdots x_k \ \ \forall \ \mathrm{Var}(t_1, t_2, R_1) \ \ \mathrm{Def}(t_1, t_2, R_1) \rightarrow [t_1 \approx t_2 \vee R_1]$$

and

$$\forall x_1 \cdots x_k \ \ \forall \ \mathrm{Var}(R_2) \ \ \mathrm{Def}(R_2) \rightarrow [R_2]$$

imply

$$\forall x_1 \cdots x_k \ \ \forall \ \mathrm{Var}(t_1, t_2, R_1, R_2, R_2') \ \mathrm{Def}(t_1, t_2, R_1, R_2, R_2') \rightarrow [R_1 \vee R_2' \ ].$$

Given Σ, one can define $[\Sigma]$ as in Definition 13. Consider the proof given in Figure 3.

The proof steps can be justified as follows:

S1 Instantiation of the first premisse.
S2 Instantiation of the second premisse.
S3 Instantiation of S2, using $[\Sigma]$.
S4 We show that the atoms in $\mathrm{Def}(R_2) \cdot [\Sigma]$ are provable. Let $A \in \mathrm{Def}(R_2)$. One can write $A = R_f(\ [w_1], \ldots, [w_n],\ [\ f(w_1, \ldots, w_n)\ ]\ )$, where $f \in \mathcal{F}_{\mathrm{Repl}}$ and $f(w_1, \ldots, w_n)$ occurs in R_2.
By definition of extension, $f(w_1, \ldots, w_n) \cdot \Sigma$ has form $f(v_1, \ldots, v_n)$, with $\mathcal{E}(w_i \cdot \Sigma,\ v_i)$, for $1 \leq i \leq n$. From Theorem 5, it follows that $f(w_1, \ldots, w_n) \cdot \Sigma$ occurs in either $R_2 \cdot \Sigma,\ \ t_1$ or t_2.
If $f(w_1, \ldots, w_n) \cdot \Sigma$ occurs in $R_2 \cdot \Sigma$, but not in R_2', then one can apply an argument, similar to the last part of the proof of Theorem 5.
((Because $\mathcal{E}(R_2 \cdot \Sigma,\ R_2')$, it is possible to rewrite $R_2 \cdot \Sigma$ into R_2', using the equality $t_1 \approx t_2$. In the rewrite sequence, there is a last term that still contains $f(v_1, \ldots, v_n)$. Because replacing in or at a v_i is not allowed, and rewriting by $t_1 \approx t_2$ removes $f(v_1, \ldots, v_n)$, either t_1 or t_2 must contain $f(v_1, \ldots, v_n) = f(w_1, \ldots, w_n) \cdot \Sigma$))
Because $f(v_1, \ldots, v_n)$ occurs in t_1, t_2 or R_2', it must be the case that

$$R_f(\ [v_1], \ldots, [v_n],\ [\ f(v_1, \ldots, v_n)\ ]\ ) \in \mathrm{Def}(t_1, t_2, R_1, R_2, R_2'). \qquad (1)$$

Since $[\Sigma]$ is a substitution, $R_f(\ [w_1], \ldots, [w_n],\ [\ f(w_1, \ldots, w_n)\ ]\ ) \cdot [\Sigma]$ equals

$$R_f(\ [w_1] \cdot [\Sigma], \ldots, [w_n] \cdot [\Sigma],\ [f(w_1, \ldots, w_n)] \cdot [\Sigma]\ ),$$

which, by Theorem 3, equals

$$R_f(\ [w_1 \cdot \Sigma], \ldots, [w_n \cdot \Sigma],\ [\ f(w_1, \ldots, w_n) \cdot \Sigma\ ]\ ). \qquad (2)$$

Since for each i, (with$1 \leq i \leq n$), $\mathcal{E}(w_i \cdot \Sigma,\ v_i)$, it follows from Theorem 4, that $[t_1] \approx [t_2] \vdash [w_i \cdot \Sigma] \approx [v_i]$. Then it follows from (1) that (2) is provable.
S5 Follows from Theorem 3.
S6 Follows from Theorem 4, because $\mathcal{E}(R_2',\ R_2 \cdot \Sigma)$.
S7 $\vee$-introduction.
S8 $\vee$-introduction.
S9 $\vee$-elimination.

Fig. 3. Proof for Non-Separating Paramodulation

	$\forall x_1 \cdots x_k \ \forall \ \mathrm{Var}(t_1, t_2, R_1) \ \mathrm{Def}(t_1, t_2, R_1) \rightarrow [t_1 \approx t_2 \vee R_1]$	(premisse)
	$\forall x_1 \cdots x_k \ \forall \ \mathrm{Var}(R_2) \ \mathrm{Def}(R_2) \rightarrow [R_2]$	(premisse)
	Fresh $x_1, \ldots, x_k$	
	Fresh $\mathrm{Var}(t_1, t_2, R_1, R_2, R_2')$	
	$\mathrm{Def}(t_1, t_2, R_1, R_2, R_2')$	
	$[t_1] \approx [t_2] \ \vee \ [R_1]$	S1
	$[t_1] \approx [t_2]$	
	$\forall \ \mathrm{Var}(R_2) \ \mathrm{Def}(R_2) \rightarrow [R_2]$	S2
	$(\ \mathrm{Def}(R_2) \rightarrow [R_2] \ ) \cdot [\Sigma]$	S3
	$[R_2] \cdot [\Sigma]$	S4
	$[R_2 \cdot \Sigma]$	S5
	$[R_2']$	S6
	$[R_1] \vee [R_2']$	S7
	$[R_1]$	
	$[R_1] \vee [R_2']$	S8
	$[R_1] \vee [R_2']$	S9
	$\forall x_1 \cdots x_k \ \forall \ \mathrm{Var}(t_1, t_2, R_1, R_2, R_2') \ \mathrm{Def}(t_1, t_2, R_1, R_2, R_2') \rightarrow [R_1 \vee R_2']$	

4 Translation of the CNF-transformation

In the previous section we have shown that it is possible to replace function symbols by arbitrary serial relations in resolution proofs on the clause level. This is possible on the condition that paramodulation steps are non-separating.

In order to be able to use resolution on unrestricted first-order formulas, one needs to tranform a first-order formula into clausal normal form (CNF). During the CNF-transformation, one normally introduces Skolem functions for existentially quantified variables, and one usually replaces some subformulas by new predicate symbols, for reasons of efficiency. (See [3], [2], [17], [9])

In this section we show that in the CNF-transformation, one can introduce serial relations instead of Skolem functions. After that, the transformation to CNF can be continued as with Skolem functions and one obtains the same clauses that one would have obtained otherwise, but with serial relations instead of Skolem functions.

The resulting CNF-transformation lies completely within first-order logic, and has a size that is polynomial in the size of the original CNF-reduction sequence.

We will consider CNF-transformations with the following general pattern: First, new names are introduced for certain subformulas that would cause exponential blow-up. After that, the formula is transformed into negation normal form. Then antiprenexing is applied on the formula. After that, the resulting formula is Skolemized. Here we will introduce serial relations instead of Skolem functions. Although the intuition behind the relation-introduction is straightforward, the actual transformation is technically involved, due to technical difficulties that we will explain shortly. After Skolemization, the resulting formula can be factored into clauses as usual.

We now explain the basic idea of the relation introduction, and after that the source of the technical difficulties. Consider the formula $\forall x\ p(x) \rightarrow \exists y\ q(x,y)$. Its Skolemization equals $\forall x\ p(x) \rightarrow q(x, f(x))$. Instead of Skolemizing, one can introduce the relation $F(x,y) := (\exists z\ q(x,z)) \rightarrow q(x,y)$ and construct the formula $\forall x\ p(x) \rightarrow \forall y\ F(x,y) \rightarrow q(x,y)$.

Relation F can be easily proven serial, because $\forall x \exists y\ (\exists z\ q(x,z)) \rightarrow q(x,y)$ is a tautology. In addition, the formula $\forall x\ p(x) \rightarrow \forall y\ ((\exists z\ q(x,z)) \rightarrow q(x,y)) \rightarrow q(x,y)$ is easily provable from $\forall x\ p(x) \rightarrow \exists y\ q(x,y)$.

Once the Skolemized formula has been replaced by its relational counterpart, the CNF-transformation can proceed in the same way as on the Skolemized formulas. However, there is one technical difficulty that is caused by the fact that standard outermost Skolemization cannot be iterated. In order to obtain Skolem terms that are as small as possible, Skolemization is usually done from outside to inside, because otherwise one would obtain nested Skolem terms.

If we replace an outermost Skolemization sequence by an outermost relation introduction sequence, then the existential variables do not disappear, but are replaced by universal quantifiers. Later Skolemization steps will depend on these universal quantifiers, and this introduces unwanted dependencies. The following example shows the problem:

Example 9. The following two formulas show the problem with outermost Skolemization. Outermost Skolemization of $\forall x\ \exists y_1\ \exists y_2\ p(x, y_1, y_2)$ results in $\forall x\ \exists y_2\ \ p(x, f_1(x), y_2)$. Skolemizing one more time results in $\forall x\ p(x, f_1(x), f_2(x))$. It appears that there exist no binary relations F_1, F_2 for which the formulas

$$\forall x\ \forall y_1\ \forall y_2\ F_1(x, y_1) \rightarrow F_2(x, y_2) \rightarrow p(x, y_1, y_2),$$
$$\forall x\ \exists y_1\ F_1(x, y_1),$$
$$\forall x\ \exists y_2\ F_2(x, y_2),$$

are provable. The problem is due to the fact that in the original formula, the y_2 can only be chosen with knowledge of y_1. The same problem appears in the formula $\forall x_1\ \exists y_1\ \forall x_2\ \exists y_2\ p(x_1, y_1, x_2, y_2)$. Outermost Skolemization results in $\forall x_1\ \forall x_2\ p(x_1, f_1(x_1), x_2, f_2(x_1, x_2)\ )$.

Again, there seems to be no way of finding relations that are serial and for which $\forall x_1\ \forall y_1\ \forall x_2\ \forall y_2\ F_1(x_1, y_1) \rightarrow F_2(x_1, x_2, y_2) \rightarrow p(x_1, y_1, x_2, y_2)$ is provable.

The problem can be solved by using innermost Skolemization instead of outermost Skolemization. Innermost Skolemization was considered in [16] and proven sound there. Innermost Skolemization proceeds in the same way as standard Skolemization, but it starts with an innermost existential quantifier, instead of an outermost existential quantifier.

Example 10. On the first formula of the previous example, one step of innermost Skolemization results in $\forall x\ \exists y_1\ p(x, y_1, f(x, y_1))$. One more step produces $\forall x\ p(x, f_1(x), f_2(x, f_1(x_1))\ )$.

Similarly, innermost Skolemization iterated on the second formula produces the final formula $\forall x_1\ \forall x_3\ p(x_1, f_1(x_1), x_2, f_2(x_1, f_1(x_1), x_2)\ )$.

Innermost Skolemization is not suitable for proof search because it results in bigger Skolem terms. However, the proof length of a resolution proof does not increase if one uses innermost Skolemization instead of outermost Skolemization. This is due to the fact that, although Skolem terms obtained from innermost Skolemization are bigger, they do not depend on more variables. As a consequence, whenever in a clause two (outermost) Skolem terms are equal, their innermost counterparts are also equal. Therefore, one can pass the clauses obtained from the outermost Skolemization to the theorem prover. If the prover returns a proof, then one can convert it into a proof of the clauses obtained from the innermost Skolemization, without improving the number of proof steps. Then this proof can be used to replace the function introductions by relation introductions. There is one remaining problem, which is caused by the fact that non-separating paramodulation steps on Skolem terms obtained from outermost Skolemization are not necessarily non-separating paramodulation steps on the corresponding Skolem terms that one obtains from innermost Skolemization. We will discuss this in Section 4.1.

Definition 20. *A formula F is in* negation normal form *if it does not contain $\leftrightarrow$ and $\rightarrow$, and every occurrence of $\neg$ is applied to an atom.*

In the rest of this paper, we assume that all first-order formulas F are *standardized apart*, i.e. no variable is bound twice in F. This can be easily obtained by renaming.

Definition 21. *Let F be a formula in negation normal form. If F contains an existentially quantified subformula, then F can be written as $F[\ \exists y\ G\ ]$. Let $x_1, \ldots, x_k$ be the free variables of $\exists y\ G$ that are bound by a quantifier in F. Let f be a new function symbol, s.t.* $\mathrm{ar}(f) = k$. *Then $F[\ G \cdot \{y := f(x_1, \ldots, x_k)\}\ ]$ is a* one-step Skolemization *of $F[\ \exists y\ G\ ]$.*
Let $F = F_1, \ldots, F_m$ be a Skolemization sequence, i.e.

- *each F_{i+1} is a one-step Skolemization of F_i, which Skolemizes some subformula $\exists y_i\ G_i$.*
- *F_m has no remaining existential quantifiers.*

F_m is an outermost Skolemization *of F, if each $\exists y_i\ G_i$ is not in the scope of another existential quantifier in F_i.*
F_m is an innnermost Skolemization *of F, if no G_i contains another existential quantifier.*

We prove the claim that instead of Skolem functions, serial relations can be obtained:

Theorem 6. *Let F be a formula in negation normal form containing an existential quantifier. Write F as $F[\ \exists y\ A(x_1, \ldots, x_k, y)\ ]$, where $x_1, \ldots, x_k$ are the quantified variables that are bound in F and free in $\exists y\ A(x_1, \ldots, x_k)$. Let $F[\ A(x_1, \ldots, x_k, g(x_1, \ldots, x_k))\ ]$ be obtained by one-step Skolemization*
There is a $(k+1)$-place relation G, for which the following formulas are provable:

SER $\forall x_1 \cdots x_k\ \exists y\ G(x_1, \ldots, x_k, y)$,
SKOL $F[\ \forall y\ G(x_1, \ldots, x_k, y) \rightarrow A(x_1, \ldots, x_k, y)\ ]$.

Proof. Take $G(x_1, \ldots, x_k, y) := (\ \exists z\ A(x_1, \ldots, x_k, z)\ ) \rightarrow A(x_1, \ldots, x_k, y)$. Then SER becomes

$$\forall x_1 \cdots x_k\ \exists y\ (\ \exists z\ A(x_1, \ldots, x_k, z)\ ) \rightarrow A(x_1, \ldots, x_k, y),$$

which is a simple tautology.
We show that SKOL is logically equivalent to F. Expanding G in SKOL yields:

$$F[\ \forall y\ (\ (\exists z\ A(x_1, \ldots, x_k, z)\ ) \rightarrow A(x_1, \ldots, x_k, y)\ ) \rightarrow A(x_1, \ldots, x_k, y)\ ].$$

This is logically equivalent to

$$F[\ \forall y\ (\ \exists z\ A(x_1, \ldots, x_k, z)\ ) \wedge \neg A(x_1, \ldots, x_k, y) \vee A(x_1, \ldots, x_k, y)\ ],$$

which in turn is equivalent to

$$F[\ \forall y\ (\ \exists z\ A(x_1, \ldots, x_k, z)\ ) \vee A(x_1, \ldots, x_k, y)\ ].$$

This final formula is equivalent to

$$F[\ \exists z\ A(x_1, \ldots, x_k, z)\ ].$$

If one replaces an innermost Skolemization sequence by an innermost relation-introduction sequence, then one obtains a formula that has the same structure as the Skolemized formula, but with the Skolem terms replaced by variables based on some function replacement [ ]. In addition, it contains definitions of form $G(x_1, \ldots, x_k, y) \rightarrow A$ that introduce the variables that are used in the translations of the Skolem functions. The fact that in the orginal formula y was in the scope of an existential quantifier $\exists y$, ensures that in the relational translation, y is in the scope of a definition $\forall y\ G(x_1, \ldots, x_k, y) \rightarrow A$. The following definition specifies more precisely the relation between the innermost Skolemization and the innermost relation-introduction.

Definition 22. *Let* [ ] *be a function replacement, replacing functions from* $\mathcal{F}_{\mathrm{Repl}}$ *and introducing variables from* $\mathcal{F}_{\mathrm{Def}}$*. Let* F *be a first-order formula that is standardized apart. A* [ ]-translation *of* F *is a formula that can be obtained as follows from* F *: Replace each atom* A *in* F *by* $[A]$*. Iteratively, insert definitions into* F *in the following way: Select some subformula* G *of* F *and some term* $f(t_1, \ldots, t_n)$ *with* $f \in \mathcal{F}_{\mathrm{Repl}}$*. Let* $\alpha = [f(t_1, \ldots, t_n)]$*. Then replace* $F[G]$ *by* $F[\ \forall \alpha\ R_f(\ [t_1], \ldots, [t_n], \alpha\ ) \rightarrow G\ ]$*. Call the result of the replacements* F'*. Then* F' *has to meet the following conditions:*

- *No variable* $\alpha \in \mathcal{F}_{\mathrm{Def}}$ *has more than one definition in* F'*.*
- *For every definition* $\forall \alpha\ R_f([t_1], \ldots, [t_n], \alpha) \rightarrow G'$ *in* F'*, the variable* α *occurs somewhere in* G'*.*
- *Let* t *be a term occurring in* F *in the scope of quantifiers* $\forall x_1 \cdots \forall x_k$*. Write* $\mathrm{Var}(t) = \{\alpha_1, \ldots, \alpha_m\}$*. Then the corresponding position in* F' *contains* $[t]$*, and the path from the root of* F' *towards the occurrence of* $[t]$ *contains all quantifiers* $\forall x_1 \cdots \forall x_k$*, all quantifiers* $\forall \alpha_1 \cdots \forall \alpha_m$ *and all definitions of* $\mathrm{Def}(t)$ *in a correct order. (No variable occurs before it is quantified)*

Theorem 7. *Let* F_1 *be obtained from* F *by innermost Skolemization. Let* F_2 *be obtained from* F *by making the corresponding relation replacements. Then there is a function replacement* [ ]*, s.t.* F_2 *is a* [ ]*-translation of* F_1*.*

Example 11. We demonstrate relation introduction on the formula $\forall x_1\ \exists y_1\ \forall x_2\ \exists y_2\ \ p(x_1, y_1, x_2, y_2)$. One step of innermost Skolemization results in $\forall x_1\ \exists y_1\ \forall x_2\ \ p(x_1, y_1, x_2, f_2(x_1, y_1, x_2))$, and one more step of innermost Skolemization yields $\forall x_1\ \forall x_2\ p(x_1, f_1(x_1), x_2, f_2(x_1, f_1(x_1), x_2))$. First put $R_{f_2}(x_1, y_1, x_2, y_2) := (\exists z\ \ p(x_1, y_1, x_2, z)\ ) \rightarrow p(x_1, y_1, x_2, y_2)$. Then one can prove

$$\forall x_1\ \exists y_1\ \forall x_2\ \forall y_2\ R_{f_2}(x_1, y_1, x_2, y_2) \rightarrow p(x_1, y_1, x_2, y_2).$$

After that, put

$$R_{f_1}(x_1, y_1) := (\ \exists z\ (\forall x_2\ \forall y_2\ R_{f_2}(x_1, z, x_2, y_2) \rightarrow p(x_1, z, x_2, y_2)\ )) \rightarrow$$

$$(\ \forall x_2\ \forall y_2\ R_{f_2}(x_1, y_1, x_2, y_2) \rightarrow p(x_1, y_1, x_2, y_2)\ ).$$

Then $\forall x_1\ \forall y_1\ R_{f_1}(x_1, y_1) \rightarrow \forall x_2\ \forall y_2\ R_{f_2}(x_1, y_1, x_2, y_2) \rightarrow p(x_1, y_1, x_2, y_2)$ is provable, together with the seriality axioms for R_{f_1} and R_{f_2}.

4.1 Replacing Innermost Skolemization by Outermost Skolemization

We now justify the claim made in the previous section that innermost Skolemization does not increase the proof length. Before we do this, we give an example that makes clear that one has to use a form of paramodulation that is more restricted than non-separating paramodulation.

Example 12. Consider the formula $\forall x\ \exists y_1\ y_2\ p(x, y_1, y_2)$ and the two clauses

$$\begin{array}{ll} C_1 & p(0, f_1(0), f_2(0)), \\ C_2 & p(0, f_1(0), f_2(0, f_1(0))). \end{array}$$

The clause C_1 is obtained by instantiating the outermost Skolemization of F by 0. The clause C_2 is obtained by making the same instantiation in the innermost Skolemization. From C_1, it is possible to derive $p(\ 0, f_1(1), f_2(0)\ )$ by non-separating paramodulation from equation $0 \approx 1$. However, it is not possible to derive $p(\ 0, f_1(1), f_2(0, f_1(0)\ )$ from C_2 by non-separating paramodulation.

The example shows that non-separating paramodulation steps on clauses with innermost Skolem terms can in general not be translated into non-separating paramodulation steps on the corresponding clauses with outermost Skolem terms. Therefore, the following more restricted version of paramodulation is needed.

Definition 23. *Let* [ ] *be a function replacement replacing functions from* $\mathcal{F}_{\mathrm{Repl}}$. *The* $\mathcal{F}_{\mathrm{Repl}}$-simultaneous paramodulation rule *is the following rule:*

$$\frac{\forall x_1 \cdots x_k \quad t_1 \approx t_2 \vee R_1 \qquad\qquad \forall x_1 \cdots x_k \quad R_2}{\forall x_1 \cdots x_k \quad R_1 \vee R_2'}$$

where R_2' *is obtained from* R_2 *by replacing arbitrary occurrences of* t_1 *by* t_2. *If at least one occurrence of* t_1 *in the scope of a function symbol from* $\mathcal{F}_{\mathrm{Repl}}$ *is replaced, then all occurrences of* t_1 *that are in the scope of a function symbol from* $\mathcal{F}_{\mathrm{Repl}}$ *have to be replaced.*

Example 13. The paramodulation step from C_1 in Example 12 was not $\mathcal{F}_{\mathrm{Repl}}$-simultaneous, because $\mathcal{F}_{\mathrm{Repl}} = \{f_1, f_2\}$. The clauses that can be obtained by $\mathcal{F}_{\mathrm{Repl}}$-simultaneous paramodulation from $p(\ 0, f_1(0), f_2(0)\ )$ are $p(\ 1, f_1(0), f_2(0)\ )$, $p(\ 0, f_1(1), f_2(1)\ )$, $\ p(\ 1, f_1(1), f_2(1)\ )$.

Using $\mathcal{F}$-simultaneous paramodulation, we can state the main result of this paper:

Theorem 8. *Given a first-order formula* F, *and a resolution refutation for which*

- *the CNF-transformation uses subformula replacement, antiprenexing and outermost Skolemization,*

- *the resolution refutation on the clause level uses the standard resolution rules but $\mathcal{F}_{\text{Repl}}$-simultaneous paramodulation, where $\mathcal{F}_{\text{Repl}}$ are the Skolem functions introduced in the CNF-transformation,*

then one can effectively obtain a purely first-order refutation of F that is polynomial in the size of the CNF-tranformation plus the size of the resolution refutation.

The proof will follow in the rest of this section. Observe that, in case one does not make any replacements at all inside Skolem functions, which is known to be complete because of the results in [5], one automatically uses $\mathcal{F}_{\text{Skol}}$-simultaneous paramodulation.

In that case, one also gets *the splitting rule* for free. The splitting rule is the following rule, used on the clause level by resolution theorem provers: If the prover derives a clause $\forall x_1 \cdots x_k \ (R_1 \vee R_2)$, s.t. no x_i occurs in both R_1 and R_2, then $\forall x_1 \cdots x_k \ (R_1 \vee R_2)$ implies $(\forall x_1 \cdots x_k \ R_1) \vee (\forall x_1 \cdots x_k \ R_2)$. These two clauses can be refuted seperately by backtracking.

In general, if some clause $\forall x_1 \cdots x_k \ R_1 \vee R_2$ is splittable, its translation $\forall x_1 \cdots x_k \ \forall \ \text{Var}(R_1, R_2) \ \text{Def}(R_1, R_2) \rightarrow [R_1] \vee [R_2]$ need not be splittable, consider for example the ground clause $p(f(0)) \vee q(f(0))$ with translation $\forall \alpha \ F(0, \alpha) \rightarrow p(\alpha) \vee q(\alpha)$. However, if one knows that one never paramodulates inside Skolem functions, one can 'glue' $q(f(0))$ to the clauses in the refutation of $p(f(0))$, without ever being forced to modify $p(f(0))$ (and have to admit that the remark about general splitting in the conclusion of [12] was incorrect)
We now continue by showing that the difference between innermost and outermost Skolemization is smaller than it appears to be. Skolem terms obtained from innermost Skolemization have exactly the same variables as Skolem terms obtained from outermost Skolemization.

Theorem 9. *Let F be a formula. Let F_1 be its outermost Skolemization. Let F_2 be its innermost Skolemization. Then F_1 and F_2 have the same logical structure (this means that they only differ inside some atoms), and each Skolem term in F_1 depends on exactly the same variables as its counterpart in F_2.*

Proof. It is easy to see that F_1 and F_2 have the same logical structure, because Skolemization does not change the logical structure, except for the elimination of existential quantifiers. We will show that the Skolem terms in F_1 and F_2 depend on the same set of variables in the following sequence of definitions and lemmas.

Definition 24. *Let F be a first-order formula. Let x and y be two variables that are quantified inside F. Let A be the subformula of F that is quantified by x. We say that y* eventually depends on x *if there exists a sequence $\exists y_1 \ B_1, \ldots, \exists y_n \ B_n$ of subformulas of F, s.t.*

1. *$\exists y_1 \ B_1$, is inside A and x is free in $\exists y_1 \ B_1$,*
2. *each $\exists y_{i+1} \ B_{i+1}$ is inside B_i and y_i is free in $\exists y_{i+1} \ B_{i+1}$, and*
3. *$y_n = y$.*

Similarly, we say that a term t eventually depends on *x in F if there exists a sequence $\exists y_1\ B_1, \ldots, \exists y_n\ B_n$ of subformulas of F, s.t.*

1. *$\exists y_1\ B_1$, is inside A and x is free in $\exists y_1\ B_1$,*
2. *each $\exists y_{i+1}\ B_{i+1}$ is inside B_i and y_i is free in $\exists y_{i+1}\ B_{i+1}$, and*
3. *y_n occurs in t.*

Lemma 2. *Let $F_1, \ldots, F_m$ be a Skolemization sequence in which each F_{i+1} is obtained from F_i by Skolemization at an arbitrary position.*

Let x be a universally quantified variable, occurring in F_1. Let y be a universally quantified variable, occurring in F_1. Let t be its Skolem term in F_m. Then x occurs in t iff y eventually depends on x in F_1.

Proof. It follows, by n times applying Lemma 3, that y eventually depends on x in F_1 iff its Skolem term t eventually depends on x in F_2. Because F_2 does not contain existential quantifiers, this is case iff x occurs in t.

Lemma 3. *Let F' be obtained from F by one-step Skolemization. Let x be a quantified variable, which occurs in both F and F'. Let A be the subformula quantified by x in F. Let A' be the subformula quantified by x in F'. Then:*

1. *Some term t in F eventually depends on x iff its counterpart t' in F' eventually depends on x.*
2. *Let y be an existentially quantified variable in F, which is not Skolemized in F'. Then y eventually depends on x in F iff y eventually depends on x in F'.*
3. *Let y be the existentially quantified variable in F which is Skolemized in F'. Then y eventually depends on x in F iff the Skolem term for y eventually depends on x in F'.*

Proof. In order to prove the 3 properties from left to right, it is sufficient to observe the following two points:

1. Let z_1 be a quantified variable occurring in F. Let G_1 be the subformula of F that is quantified by z_1. Let $\exists z_2\ G_2$ and $\exists z_3\ G_3$ be subformulas of F, s.t.
   - $\exists z_2\ G_2$ is a subformula of G_1 and z_1 is free in G_2,
   - $\exists z_3\ G_3$ is a subformula of G_2 and z_2 is free in G_3.

   If F' is obtained from F' by Skolemizing z_2, then $\exists z_3\ G_3'$ is a subformula of G_1', and z_1 is free in $\exists z_3\ G_3'$.
2. Let z_1 be a quantified variable occurring in F. Let G_1 be the subformula of F that is quantified by z_1. Let $\exists z_2\ G_2$ be a subformula of F, let t be a term occurring in G_2, s.t.
   - $\exists z_2\ G_2$ is a subformula of G_1 and z_1 is free in $\exists z_2\ G_2$,
   - the variable z_2 occurs in t.

   If F' is obtained from F' by Skolemizing z_2, then z_1 occurs in t'.

Using this, the 3 properties can be easily proven from left to right.

Next we prove the 3 properties from right to left. In order to prove these, we need the following 2 properties, which are essentially the converses of the properties above.

1. Let z_1 be a quantified variable occurring in F. Let G_1 be the subformula of F that is quantified by z_1. Let $\exists z_3\ G_3$ be a subformula of G_1. Assume that both z_1 and z_3 are not Skolemized in F'. Then G_1' is a subformula of F' and still quantified by z_1. Also $\exists z_3\ G_3'$ remains a subformula of G_1'. Assume that z_1 is free in $\exists z_3\ G_3'$. Then either
   - z_1 is free in $\exists z_3\ G_3$, or
   - there is an existentially quantified subformula $\exists z_2\ G_2$ of F, s.t.
     - $\exists z_2\ G_2$ is a subformula of G_1 and z_1 is free in $\exists z_2\ G_2$,
     - $\exists z_3\ G_3$ is a subformula of G_2 and z_2 is free in G_3.

   Assume that the first case does not hold. Then z_1 occurs in the Skolem term introduced in F'. Let $\exists z_2\ G_2$ be the corresponding subformula of F. Because z_1 occurs in the Skolem term, it must be the case that z_1 is free in $\exists z_2\ G_2$. As a consequence, $\exists z_2\ G_2$ is a subformula of G_1. Because the Skolem term occurs in G_3', it must be the case that G_2 overlaps with G_3. Then either $\exists z_2\ G_2$ occurs in G_3, or $\exists z_3\ G_3$ occurs in G_2. The first possibility cannot happen, because in that case z_1 would also occur in G_3.
2. Let z_1 be a quantified variable occurring in F. Let G_1 be the subformula of F that is quantified by z_1. Let t be a term occurring in G_1. Assume that z_1 is not Skolemized in F'. Assume that z_1 occurs in t'. Then either
   - z_1 occurs in t, or
   - there exists an existentially quantified subformula $\exists z_2\ G_2$ of F, s.t.
     - $\exists z_2\ G_2$ is a subformula of G_1 and z_1 is free in $\exists z_2\ G_2$,
     - z_2 is free in t.

We now have shown that terms obtained from innermost Skolemization do not contain more variables than terms obtained from outermost Skolemization. In order to prove that resolution refutations from sets of clauses with innermost Skolem terms can be translated into resolution refutations from sets of clauses with outermost Skolem terms, we also need to look into their structure:

Definition 25. *We recursively define the set of* Skolem-type *terms.*

- *A variable x is also a Skolem-type term.*
- *If $t_1, \ldots, t_n$ are variables or Skolem-type terms, f is a Skolem function, then $f(t_1, \ldots, t_n)$ Skolem-type term.*

Given a Skolem-type term $f(t_1, \ldots, t_n)$, we call the positions of the t_i that contain other Skolem-type terms internal positions. *The positions that contain the variables are called* external. *An outer-inner transformation Θ is a function that assigns to Skolem terms of form $f(x_1, \ldots, x_n)$ Skolem-type terms. Θ must satisfy the following conditions:*

1. *$\Theta(\ f(x_1, \ldots, x_n)\ )$ is a Skolem-type term containg exactly variables $x_1, \ldots, x_n$ and some additional Skolem functions. In particular, there are no non-Skolem functions in $\Theta(\ f(x_1, \ldots, x_n)\ )$.*
2. *If some Skolem function g occurs in $\Theta(\ f(x_1, \ldots, x_n)\ )$, and its i-th argument is an internal position, then its i-th argument is an internal position in every $\Theta(\ f'(x_1, \ldots, x_m)\ )$ in which g occurs.*

Outer-inner transformation Θ is extended to terms, literals and clauses as expected, by recursion.

Theorem 10. *Let $\mathcal{F}_{\mathrm{Repl}}$ be the set of Skolem functions. Let Θ be an outer-to-inner transformation. Let $C_1, \ldots, C_n$ be some set of clauses. If $C_1, \ldots, C_n$ have a resolution refutation using $\mathcal{F}$-non-separating paramodulation, then $\Theta(C_1), \ldots, \Theta(C_n)$ have a resolution refutation using $\mathcal{F}$-simultaneous paramodulation.*

Proof. It is not hard to see that all for all (unrestricted) resolution steps holds: If D can be obtained from $D_1, \ldots, D_k$, (with $k = 1$ or $k = 2$), then $\Theta(D)$ can be obtained from $\Theta(D_1), \ldots, \Theta(D_k)$. In addition one has to show that if D is obtained from D_1 and D_2 by $\mathcal{F}_{\mathrm{Repl}}$-simultaneous paramodulation, then $\Theta(D)$ can be obtained from $\Theta(D_1)$ and $\Theta(D_2)$ by $\mathcal{F}$-simultaneous paramodulation. Write

$$D_1 = \forall x_1 \cdots x_k \; t_1 \approx t_2 \vee R_1, \qquad D_2 = \forall x_1 \cdots x_k \; R_2,$$
$$D = \forall x_1 \cdots x_k \; R_1 \vee R_2',$$

where R_2' is obtained from R_2 by $\mathcal{F}_{\mathrm{Repl}}$-simultaneous paramodulation. Then

$$\Theta(D_1) = \forall x_1 \cdots x_k \; \Theta(t_1) \approx \Theta(t_2) \vee \Theta(R_1), \qquad \Theta(D_2) = \forall x_1 \cdots x_k \; \Theta(R_2),$$

and

$$\Theta(D) = \forall x_1 \cdots x_k \; \Theta(R_1) \vee \Theta(R_2').$$

Let $\mathcal{E} = \{ \; \Sigma(t_1) \approx \Sigma(t_2) \; \}$. We need to show that there exists an extension Σ of $\mathcal{E}$, s.t. $\mathcal{E}(\; \Theta(R_2) \cdot \Sigma, \; \Theta(R_2') \;)$.

If t_1 is not replaced inside Skolem terms in R_2, then one can define Σ from $f(w_1, \ldots, w_n) \cdot \Sigma = f(w_1, \ldots, w_n)$, for all Skolem terms $f(w_1, \ldots, w_n)$.

Otherwise, Θ is defined as follows: For each $f \in \mathcal{F}_{\mathrm{Repl}}$, put $f(w_1, \ldots, w_n) \cdot \Sigma = f(v_1, \ldots, v_n)$, where $w_i = v_i$ if w_i is on an internal position of f. Otherwise v_i is obtained from w_i by replacing all occurrences of $\Sigma(t_1)$ that are not in the scope of a function from $\mathcal{F}_{\mathrm{Repl}}$ by $\Sigma(t_2)$.

4.2 The Complete Transformation

We have now discussed the technical difficulties, and we are ready to describe the complete proof transformation.

The CNF-transformation usually starts with *subformula replacement*, in order to avoid exponential blowup later in the transformation, see [17], [2], [9], [11]. For example, the following formula will result in 2^p clauses, when naively factored into clausal normal form: $(a_1 \wedge b_1) \vee \cdots \vee (a_p \wedge b_p)$.

Definition 26. *Let F be a first-order formula. A* formula definition (relative to) *F is a formula of form*

$$\forall x_1 \cdots x_k \; A(x_1, \ldots, x_k) \leftrightarrow R(x_1, \ldots, x_k),$$

in which A is a k-ary propositional symbol which does not occur in F, and R is a k-ary relation.

The following is standard:

Theorem 11. *Suppose that there exists a proof Π of*

$$F \wedge \quad \forall x_1 \cdots x_k \; A(x_1, \ldots, x_k) \leftrightarrow R(x_1, \ldots, x_k) \vdash \bot,$$

then there exists a proof of $F \vdash \bot$.

Proof. First substitute $\Pi' := \Pi[A := R]$. The result Π' is a proof of

$$F \wedge \quad \forall x_1 \cdots x_k \; R(x_1, \ldots, x_k) \leftrightarrow R(x_1, \ldots, x_k) \vdash \bot.$$

Because $\forall x_1 \cdots x_k \; R(x_1, \ldots, x_k) \leftrightarrow R(x_1, \ldots, x_k)$ is a tautology, one can obtain a proof Π'' of $F \vdash \bot$.

In the example before Definition 26, one can replace the formula by $c_1 \leftrightarrow (a_1 \wedge b_1), \ldots, c_p \leftrightarrow (a_p \wedge b_p), \;\; c_1, \ldots, c_p$, which can be easily factored into a clausal normal form of size $4p$.
After replacement of subformulas, the formula is transformed into negation normal form. After that, usually *antiprenexing* (also called *miniscoping*) is attempted, see [2]. Antiprenexing tries to reduce the scope of quantifiers using transformations of form $(Qx \; P(x) \wedge Q) \Rightarrow (Qx \; P(x)) \wedge Q$, in case x is not free in Q. Such transformations are sound, provably correct in first-order logic, and they sometimes reduces the dependencies between quantifiers, which may result in smaller Skolem terms. As an example, consider the formula $\forall x \; (\; P(x) \rightarrow \exists y \; Q(y) \;)$. Since these transformations take place before Skolemization, they are not affected by our proof transformation, and they can be carried out as usual.
After Skolemization, the resulting formula is factored into clauses through the following procedure:

Definition 27. *Let F be a first-order formula in negation normal form, that is standardized apart, and which does not contain existential quantifiers.*

1. *Replace each conjunction $A \wedge B$ by one of A or B.*
2. *Move universal quantifiers forward, using rules*

$$(\; \forall x \; A \;) \vee B \Rightarrow \forall x \; (A \vee B), \; \textit{and} \; A \vee (\; \forall x \; B \;) \Rightarrow \forall x \; (A \vee B).$$

3. *Replace universal quantifiers $\forall x \; A$, for which x is not free in A, by A.*

The different clauses are obtained by making different choices in Step 1.

Theorem 12. *Let F be a first-order formula in negation normal form, that is standardized apart, and which does not contain any existential quantifiers. Let $[\;]$ be a function replacement, and let F' be a translation of F. (See Definition 22) Then, for every clause $\forall x_1 \cdots x_k \; R$ that can be obtained from F using the factoring procedure of Definition 27 there is a clause $\forall x_1 \cdots x_k \; \forall \; Var(R) \rightarrow \mathrm{Def}(R) \rightarrow [R]$, which can be obtained from F' by adding the following rules to Definition 27.*

2a

$$(\ \forall\alpha\ R_f(t_1,\ldots,t_n,\alpha\ ) \rightarrow A\ ) \vee B \Rightarrow \forall\alpha\ R_f(t_1,\ldots,t_n,\alpha) \rightarrow (A \vee B),$$

and

$$A \vee (\ \forall\alpha\ R_f(t_1,\ldots,t_n,\alpha) \rightarrow B\ ) \Rightarrow \forall\alpha\ R_f(t_1,\ldots,t_n,\alpha) \rightarrow (A \vee B).$$

3a *Replace universal quantifiers* $\forall\alpha\ R_f(t_1,\ldots,t_n,\alpha) \rightarrow A$, *for which* α *does not occur in* A *by* A. *(In order to do this, one needs the seriality axiom for* R_f*)*

5 Conclusions and Future Work

We gave a method for translating resolution proofs that include the CNF-transformation into purely first-order proofs. The method is efficient and structure-preserving. On the clause level, the resolution prover can make use of all of the standard resolution rules, but paramodulation has to be restricted. The CNF-transformer can make use of subformula replacement and standard Skolemization.

Paramodulation has to be carried out in such a way that all occurrences of the replaced term inside Skolem functions are treated consistently. Either they are all replaced, or none of them is replaced. A common refinement of paramodulation, in which no replacements at all are made inside Skolem terms, and which is known to be complete, is covered by this restriction. In that case, one can also keep the splitting rule.

We intend to implement the proof generation method, and see how well the method performs in practice. It needs to be seen how readable the resulting proofs are. In many cases, Skolem functions are meaningful (for example the Skolem function for $\exists y$ in the power set axiom $\forall x\ \exists y\ \forall \alpha\ (\ \alpha \subseteq x \leftrightarrow \alpha \in y)\ \ )$ and such Skolem functions are better not eliminated. One explanation could be that for such cases, functionality of the Skolem function is provable within the theory.

When translating resolution proofs on the clause level, there are quite some variations possible, which are not yet fully explored. As an example, if both f and g are Skolem functions, then one can obtain different variables for the two occurrences of $g(x)$ in the resolvent of $\forall x\ p(f(x)) \vee p(g(x))$ with $\forall x\ \neg p(f(x)) \vee q(g(x))$. In some cases, it would be possible to obtain more liberal paramodulation in this way.

In addition, there are some variations possible when generating the serial relations during CNF-transformation. Currently, we use the weakest possible such relations.

Finally, it needs to be checked if the reduction from improved Skolemization to standard Skolemization, that was described in [10] can be combined with the proof generation method of this paper.

6 Acknowledgements

It was due to a question asked by George Moser during the Deduktionstreffen 2002 in Freiburg, that I started thinking about the problem of eliminating Skolemization, but allowing the cut rule in proofs. I thank Marc Bezem for discussions on the subject of this paper and Dimitri Hendriks for the pleasant cooperation in [6], which is one of the roots of the current paper.

References

1. Jeremy Avigad. Eliminating definitions and skolem functions in first-order logic. In Harry Mairson, editor, *Proceedings of the 16-th Annual IEEE Symposion on Logic in Computer Science*, LICS, pages 139–146, Boston, Massachusetts, June 2001 2001. IEEE Computer Society.
2. Matthias Baaz, Uwe Egly, and Alexander Leitsch. Normal form transformations. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 5, pages 275–333. Elsevier Science B.V., 2001.
3. Matthias Baaz, Christian G. Fermüller, and Alexander Leitsch. A non-elementary speed-up in proof length by structural clause form transformation. In *IEEE Symposion on Logic in Computer Science 1994*, pages 213–219, 1994.
4. Matthias Baaz and Alexander Leitsch. On skolemization and proof complexity. *Fundamenta Informatika*, 4(20):353–379, 1994.
5. Leo Bachmair, Harald Ganzinger, Christopher Lynch, and Wayne Snyder. Basic paramodulation. *Information and Computation*, 121(2):172–192, 1995.
6. Marc Bezem, Dimitri Hendriks, and Hans de Nivelle. Automated proof construction in type theory using resolution. In David McAllester, editor, *Automated Deduction - CADE-17*, number 1831 in LNAI, pages 148–163. Springer Verlag, 2000.
7. Marc Bezem, Dimitri Hendriks, and Hans de Nivelle. Automated proof construction in type theory using resolution. *Journal of Automated Reasoning*, 29(3-4):253–275, December 2002.
8. Peter Clote and Jan Krajíček. *Arithmetic, Proof Theory and Computational Complexity*, volume 23 of *Oxford Logic Guides*. Oxford Science Publications, 1993.
9. Thierry Boy de la Tour. An optimality result for clause form transformation. *Journal of Symbolic Computation*, 14:283–301, 1992.
10. Hans de Nivelle. Extraction of proofs from the clausal normal form transformation. In Julian Bradfield, editor, *Proceedings of the 16 International Workshop on Computer Science Logic (CSL 2002)*, volume 2471 of *Lecture Notes in Artificial Intelligence*, pages 584–598, Edinburgh, Scotland, UK, September 2002. Springer.
11. Hans de Nivelle. Implementing the clausal normal form transformation with proof generation. In Boris Konev and Renate Schmidt, editors, *Fourth Workshop on the Implementation of Logics*, volume ULCS-03-018, pages 69–83. University of Liverpool, Department of Computer Science, September 2003.
12. Hans de Nivelle. Translation of resolution proofs into short first-order proofs without choice axioms. In Franz Baader, editor, *Automated deduction, CADE-19 : 19th International Conference on Automated Deduction*, volume 2741 of *Lecture Notes in Artificial Intelligence*, pages 365–379, Miami, USA, July 2003. Springer.
13. Xiaorong Huang. Translating machine-generated resolution proofs into ND-proofs at the assertion level. In Norman Y. Foo and Randy Goebel, editors, *Topics in Artificial Intelligence, 4th Pacific Rim International Conference on Artificial Intelligence*, volume 1114 of *LNCS*, pages 399–410. Springer Verlag, 1996.

14. William McCune and Olga Shumsky. Ivy: A preprocessor and proof checker for first-order logic. In Matt Kaufmann, Pete Manolios, and J. Moore, editors, *Using the ACL2 Theorem Prover: A tutorial Introduction and Case Studies.* Kluwer Academic Publishers, 2002? preprint: ANL/MCS-P775-0899, Argonne National Labaratory, Argonne.
15. Robert Nieuwenhuis and Albert Rubio. Paramodulation-based theorem proving. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 7, pages 371–443. Elsevier Science, B.V., 2001.
16. Andreas Nonnengart. Strong skolemization. Technical Report MPI-I-96-2-010, Max Planck Institut für Informatik Saarbrücken, 1996.
17. Andreas Nonnengart and Christoph Weidenbach. Computing small clause normal forms. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 6, pages 335–367. Elsevier Science B.V., 2001.
18. V.P. Orevkov. Lower bounds for increasing complexity of derivations after cut elimination. *Zapiski Nauchnykh Seminarov Leningradskogo Otdeleniya Matematicheskogo Instituta Imenyi V.A. Steklova AN SSSR*, 88:137–161, 1979. English translation in Journal of Soviet Mathematics 2337–2350, 1982.
19. Frank Pfenning. Analytic and non-analytic proofs. In R.E. Shostak, editor, *7th International Conference on Automated Deduction*, volume 170 of *Lecture Notes in Artificial Intelligence*, pages 394–413, Napa, California, USA, May 1984. Springer Verlag.
20. R. Statman. Lower bounds on herbrand's theorem. In *Proceedings of the American Mathematical Society*, volume 75-1, pages 104–107. American Mathematical Society, June 1979.