

Definitional Reflection and the Completion

Peter Schroeder-Heister *

Wilhelm-Schickard-Institut, Universität Tübingen
Sand 13, 72076 Tübingen, Germany
e-mail: psh@logik.informatik.uni-tuebingen.de

Abstract. The logic of definitional reflection is extended with a theory of free equality. Based on this equality theory a sequent-style notion of the completion of a definition is motivated. Definitional reflection with free equality turns out to be equivalent to the completion in this sense.

1 Introduction

The principle of definitional reflection ([9, 18]) is an extension of a sequent-style interpretation of definitional clauses for atoms, allowing one to introduce an atom on the left side of the sequent sign. For example, if in propositional logic the atom a is defined by the three clauses

$$a \leftarrow C_1 \quad a \leftarrow C_2 \quad a \leftarrow C_3 ,$$

then definitional reflection has the rule

$$\frac{\Gamma, C_1 \vdash A \quad \Gamma, C_2 \vdash A \quad \Gamma, C_3 \vdash A}{\Gamma, a \vdash A}$$

as one of its instances. It is dual to the more usual rule for introducing an atom a on the right side of the sequent sign, which has

$$\frac{\Gamma \vdash C_1}{\Gamma \vdash a} \quad \frac{\Gamma \vdash C_2}{\Gamma \vdash a} \quad \frac{\Gamma \vdash C_3}{\Gamma \vdash a}$$

as instances. Because of this symmetry we speak of \mathcal{D} -left ($\mathcal{D}\vdash$) and \mathcal{D} -right ($\vdash\mathcal{D}$) rules, respectively, where \mathcal{D} is the definition (= finite set of definitional clauses) assumed to be given. ($\vdash\mathcal{D}$) and ($\mathcal{D}\vdash$) are a powerful extension of Gentzen-style sequent systems, which were originally designed for logically compound formulas, to systems which also include rules for atoms.

In the given propositional example, definitional reflection corresponds to Clark's [3] idea of database completion. It is obvious that the above rules express, in a sequent style fashion, the axiom

$$a \leftrightarrow C_1 \vee C_2 \vee C_3,$$

* I would like to thank Roy Dyckhoff, Torkel Franzén, an anonymous reviewer and especially Robert Stärk for many helpful comments and suggestions. — This work was supported by DFG grants Schr 275/8-1 and Schr 275/11-1, and by Esprit Basic Research Working Group 7232 (GENTZEN).

which is the completed definition of a in Clark's sense. This suggests to examine in detail the exact relationship between definitional reflection with individual variables and the completion. Such an investigation would not only shed some light on definitional reflection but also on the completion. In the case of an equivalence between definitional reflection and the completion we would obtain a direct computational interpretation of the completion, in so far as definitional reflection has such an interpretation.

If variables are present, there are two variants of definitional reflection (see [18]): the local or logical version $(\mathcal{D}\vdash)$ and the global or ω -version $(\mathcal{D}\vdash)_\omega$. Given a finite set \mathcal{D} of clauses of the form $b \leftarrow C$, where b is an atom and C a formula of first-order intuitionistic logic, then they can be formulated as

$$(\mathcal{D}\vdash) \frac{\{\Gamma, C\sigma \vdash A : a = b\sigma \text{ for some } b \leftarrow C \in \mathcal{D}\}}{\Gamma, a \vdash A}$$

and

$$(\mathcal{D}\vdash)_\omega \frac{\{\Gamma\sigma, C\sigma \vdash A\sigma : \sigma = mgu(a, b) \text{ for some } b \leftarrow C \in \mathcal{D}\}}{\Gamma, a \vdash A},$$

respectively, while the right-introduction rule for atoms is

$$(\vdash\mathcal{D}) \frac{\Gamma \vdash C\sigma}{\Gamma \vdash a\sigma} \quad a \leftarrow C \in \mathcal{D}$$

in both cases. For $(\mathcal{D}\vdash)$ we have to require a certain proviso ensuring its closure under substitution; for $(\mathcal{D}\vdash)_\omega$ we just require that variables in clauses $b \leftarrow C$ are standardized apart from those in $\Gamma, a \vdash A$. It can be shown that $(\mathcal{D}\vdash)_\omega$ is equal in power to $(\mathcal{D}\vdash)_\omega + (\omega)$, where (ω) is the following rule:

$$(\omega) \frac{\{\Gamma\sigma, a\sigma \vdash A\sigma : \sigma = mgu(a, b) \text{ for some } b \leftarrow C \in \mathcal{D}\}}{\Gamma, a \vdash A}.$$

This rule expresses the idea that, in order to show that something holds, it is sufficient to show that all its closed substitution instances hold — an idea which is related to the ω -rule in arithmetic. It can be realized by a finitary rule, since to prove $\Gamma\theta, a\theta \vdash A\theta$ for any ground substitution θ we just need to prove $\Gamma\theta, a\theta \vdash A\theta$ for any substitution θ , for which $a\theta$ is defined, i.e., for which a can be inferred by a definitional clause. And all those θ are specializations of the mgus σ computed in the premisses of (ω) (see [18]). $(\mathcal{D}\vdash)_\omega$ is called a “global” rule since it justifies an assertion about an atom a by reference to all its instances. In contradistinction to that, $(\mathcal{D}\vdash)$ is local since it just refers to the way a itself rather than its instances are defined.

The rule $(\mathcal{D}\vdash)_\omega$ was proposed independently by Eriksson [5, 6] and Girard [8] in partially different contexts and under different names. As argued in [18], it has certain computational deficiencies as compared to just $(\mathcal{D}\vdash)$. However, it will turn out that it is $(\mathcal{D}\vdash)_\omega$ rather than $(\mathcal{D}\vdash)$ which corresponds to the completion.

That $(\mathcal{D}\vdash)$ does not express the idea of the completion can be seen as follows. Suppose \mathcal{D} consists of the rules

$$p(t_1) \leftarrow q \quad p(t_2) \leftarrow q$$

for different closed terms t_1 and t_2 . Then from the completion of \mathcal{D}

$$p(x) \rightarrow q$$

should follow. However, the sequent

$$p(x) \vdash q$$

cannot be derived from $(\mathcal{D}\vdash)$ (but from $(\mathcal{D}\vdash)_\omega$). Therefore in the following by “definitional reflection” we always mean the system based on $(\mathcal{D}\vdash)_\omega$ rather than $(\mathcal{D}\vdash)$.

Eriksson has presented $(\mathcal{D}\vdash)_\omega$ in such a way that it can be applied to higher order logics as long as there are finite complete sets of unifiers to be considered in $(\mathcal{D}\vdash)_\omega$ instead of mgus.¹ As usual with the completion, we confine ourselves to first-order logic. In this sense the completion is less general than the more universal principle $(\mathcal{D}\vdash)_\omega$.²

Our results are limited in the sense that they establish an equivalence between the completion of a definition and a declarative system based on definitional reflection, but do not relate it to a computational interpretation of this system. In particular, we do not relate the completion to negation as failure, which one would expect to appear, at least in some disguised form, in such a computational interpretation. This is basically due to the fact that the computational interpretation of $(\mathcal{D}\vdash)_\omega$ is not straightforward.³

The rest of this paper is divided as follows. In Section 2 we present a formal system of intuitionistic logic with definitional reflection. Section 3 extends this system with a sequent-style theory of free equality which is needed for the treatment of the completion. Basic features of this theory (such as representation of unification and invertibility of rules) are proved in some detail. In Section 4 a proof-theoretic version of the completion of a definition is defined, and in Section 5 its equivalence with the theory of definitional reflection plus free equality is established. Section 6 shows that the definition of equality by an identity clause according to Eriksson and Girard is of equal power as its characterization by the equality principles of Section 3. Section 7 relates the results obtained to substructural and other variants of the theory.

2 Intuitionistic Logic with Definitional Reflection

As our system with definitional reflection we here choose intuitionistic first-order logic enlarged with principles $(\vdash\mathcal{D})$ and $(\mathcal{D}\vdash)_\omega$ for the introduction of an atom on the right or left side of the sequent sign. We consider a language based on a fixed (possibly infinite) set of n -ary function and predicate symbols ($n \geq 0$) and infinitely many individual variables. We use x, y, z to stand for individual variables, f, g, h for functions, r, s, t for terms, p, q for predicates, a, b, c for atoms, A, B, C for formulas and Γ for multisets of formulas (all with and without primes and indices). Sequents are written as $\Gamma \vdash A$, where notations like $\Gamma_1, \Gamma_2 \vdash A$ or $A, B, \Gamma \vdash C$ are understood in the obvious way. Definitional clauses are of the form $a \Leftarrow C$. Given a definition \mathcal{D} , the

¹ Actually, he gives an even more general rule which also comprises $(\mathcal{D}\vdash)$.

² For example, definitional reflection is related to principles of pattern matching in constructive type theory (see [4]).

³ In this respect $(\mathcal{D}\vdash)_\omega$ differs from $(\mathcal{D}\vdash)$, for which a logic programming language exists (see [2]).

logic of definitional reflection over \mathcal{D} , called $\mathbf{DR}(\mathcal{D})$, is then given by the following inference rules:

$$\begin{array}{ll}
(I) \frac{}{a \vdash a} & (\perp) \frac{}{\Gamma, \perp \vdash A} \\
(\top \vdash) \frac{}{\vdash \top} & (\top \vdash) \frac{\Gamma \vdash A}{\Gamma, \top \vdash A} \\
(\wedge \vdash) \frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{\Gamma_1, \Gamma_2 \vdash A \wedge B} & (\wedge \vdash) \frac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C} \\
(\vdash \vee) \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \vee B} & (\vee \vdash) \frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma, A \vee B \vdash C} \\
(\vdash \rightarrow) \frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} & (\rightarrow \vdash) \frac{\Gamma_1 \vdash A \quad \Gamma_2, B \vdash C}{\Gamma_1, \Gamma_2, A \rightarrow B \vdash C} \\
(\vdash \forall) \frac{\Gamma \vdash A[y/x] \quad y \text{ new}}{\Gamma \vdash \forall x A} & (\forall \vdash) \frac{\Gamma, A[t/x] \vdash C}{\Gamma, \forall x A \vdash C} \\
(\vdash \exists) \frac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x A} & (\exists \vdash) \frac{\Gamma, A[y/x] \vdash C}{\Gamma, \exists x A \vdash C} \quad y \text{ new} \\
(\vdash \mathcal{D}) \frac{\Gamma \vdash C \sigma \quad a \Leftarrow C \in \mathcal{D}}{\Gamma \vdash a \sigma} & (\mathcal{D} \vdash)_\omega \frac{\{\Gamma \sigma, C \sigma \vdash A \sigma : \sigma = mgu(a, b) \text{ for some } b \Leftarrow C \in \mathcal{D}\}}{\Gamma, a \vdash A} \\
(Thin) \frac{\Gamma \vdash B}{\Gamma, A \vdash B} & (Contr) \frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B}
\end{array}$$

In applications of $(\mathcal{D} \vdash)_\omega$ we always assume that variables are standardized apart, i.e. variables in clauses are different from variables in the derivation we are constructing. In general, when using substitutions, we always assume that they are defined, i.e. that bound variables are appropriately renamed if necessary, when substitutions are applied. (This could be achieved in the standard way by the syntactic division between free and bound variables.) Obviously, the rule $(\top \vdash)$ can be obtained by $(Thin)$. However, in order to make the following proofs independent of $(Thin)$, we use it as a primitive rule. It can easily be seen that the converse of $(\top \vdash)$ is admissible, i.e., without harm we can add and delete \top on the left of the sequent sign. The rule (\perp) is derivable from $(\mathcal{D} \vdash)_\omega$ if we consider \perp a 0-ary predicate not defined in \mathcal{D} . However, it is useful to have \perp as an explicit logical constant to keep the logic without definitional clauses independent of the \mathcal{D} -rules.

Instead of having thinning and contraction as primitive rules, we could built them implicitly into the other rules in the well-known way. In particular, in order to guarantee contraction, $(\mathcal{D} \vdash)_\omega$ would have to be formulated with $\Gamma \sigma, a \sigma, C \sigma \vdash A \sigma$ rather than $\Gamma \sigma, C \sigma \vdash A \sigma$ in the premiss set. However, this way of proceeding would make comparison with substructural variants of the theory more difficult (see Section 7). The contraction-free version is particularly interesting for definitional reflection, since it permits full cut elimination, whereas the system with contraction does not (the standard counterexample uses $\mathcal{D} = \{p \Leftarrow p \rightarrow \perp\}$, see [17, 18]). By “cut” we mean the

rule

$$(Cut) \frac{\Gamma_1 \vdash A \quad \Gamma_2, A \vdash B}{\Gamma_1, \Gamma_2 \vdash B},$$

which is *not* part of $\mathbf{DR}(\mathcal{D})$.

In the following, we will sometimes write applications of $(\mathcal{D}^+)_\omega$ as

$$\frac{\{\Gamma\sigma_i, C_i\sigma_i \vdash A_i\sigma_i\}_i}{\Gamma, a \vdash A}.$$

Here $\{\dots\}_i$ expresses that for each program clause we have a premiss of the form indicated, where $\sigma_i = mgu(a, b_i)$ for the i -th program clause $b_i \Leftarrow C_i$, and where this premiss is missing if there is no such mgu. If σ is a substitution then $[t/x]\sigma$ and $\sigma[t/x]$ denote the composition of the substitution $[t/x]$ with σ and vice versa.

3 Free Equality

In order to formulate rules for the completion of a definition, we need a sequent-style theory of free equality on the domain generated by the function symbols (including individual constants) of our language. As such we use the very elegant cut-free formalization by Sahlin, Franzén & Haridi [16]. When added to $\mathbf{DR}(\mathcal{D})$, equality cuts (i.e., cuts with equations as cut formulas) can still be eliminated, although the cut rule is not admissible in general. Eliminability of equality cuts is crucial for the equivalence of definitional reflection and the completion. Therefore it is not sufficient for us just to add the axioms of Clark's [3] equality theory to $\mathbf{DR}(\mathcal{D})$.

In this section we prove central metatheoretic properties of the sequent system for free equality in some detail.⁴ Apart from the purposes of the present paper, this is justified by the general interest this system has for the proof-theoretic treatment of unification and logic programming.

We enlarge our language with a binary predicate \equiv for equality and postulate the following system \mathbf{E} of inference rules:

$$\begin{aligned} (\vdash \equiv) & \frac{}{\vdash t \equiv t} & (\equiv \vdash) & \frac{\Gamma \vdash A}{\Gamma, x \equiv x \vdash A} \\ (Repl) & \frac{\Gamma[t/x] \vdash A[t/x]}{x \equiv t, \Gamma \vdash A} & \frac{\Gamma[t/x] \vdash A[t/x]}{t \equiv x, \Gamma \vdash A} & \text{provided } x \text{ does not occur in } t \\ (\neq) & \frac{}{f(t_1, \dots, t_n) \equiv g(t'_1, \dots, t'_n), \Gamma \vdash A} & \text{provided } f \text{ and } g \text{ are distinct } (m, n \geq 0) \\ (occ) & \frac{}{x \equiv t, \Gamma \vdash A} & \frac{}{t \equiv x, \Gamma \vdash A} & \text{provided } x \text{ is a proper subterm of } t \\ (inj) & \frac{t_1 \equiv t'_1, \dots, t_n \equiv t'_n, \Gamma \vdash A}{f(t_1, \dots, t_n) \equiv f(t'_1, \dots, t'_n), \Gamma \vdash A} & (n \geq 0) \end{aligned}$$

⁴ These properties are mentioned by Sahlin *et al.* in [16] but, due to their purposes being different, not proven syntactically. - Somewhat loosely, we continue to speak about free equality as characterized by the following system. Sahlin *et al.* speak of "quasi-free" identity, since the property of being freely generated is not first-order.

Here the label “(Repl)” stands for “replacement”, “(inj)” for “injectivity” of functions, and “(occ)” should remind one of the “occurs check” in unification. $\mathbf{DRE}(\mathcal{D})$ is the system obtained from $\mathbf{DR}(\mathcal{D})$ by adding the equality theory \mathbf{E} , and by restricting (Thin) and (Contr) to the case where A is not an equality formula. The reason for this restriction is that we want to prove that thinning and contraction for equality formulas:

$$(Thin^{\equiv}) \frac{\Gamma \vdash A}{\Gamma, s \equiv t \vdash A} \quad (Contr^{\equiv}) \frac{\Gamma, s \equiv t, s \equiv t \vdash A}{\Gamma, s \equiv t \vdash A}$$

are both admissible in $\mathbf{DRE}(\mathcal{D})$ — a result, which pertains to substructural versions of the system, in which thinning and/or contraction are not available as primitive rules.

\mathbf{E} is exactly the system for equality proposed in [16], with the exception that we have formulated (\vdash^{\equiv}) without implicit thinning and have added $(\equiv\vdash)$. Although it is easy to see that the freeness axioms of Clark’s equality theory [3, p. 304] are derivable in \mathbf{E} , it is not immediately obvious that the basic principles of general equality hold, in particular the general replacement rule

$$(Repl)' \frac{\Gamma[t/x] \vdash A[t/x]}{s \equiv t, \Gamma[s/x] \vdash A[s/x]} \quad \frac{\Gamma[t/x] \vdash A[t/x]}{t \equiv s, \Gamma[s/x] \vdash A[s/x]}$$

without any restriction on t and s (see [11, 13, 15]). For that we need closure under substitution (Lemma 2 below). First we show that \mathbf{E} is a theory of first-order unification.

Lemma 1 (Unification lemma) *In \mathbf{E} the following rules are derivable:*

- (i) $\frac{\Gamma \sigma \vdash A \sigma}{s_1 \equiv t_1, \dots, s_n \equiv t_n, \Gamma \vdash A}$ if σ solves the system of equations $\{s_1 \equiv t_1, \dots, s_n \equiv t_n\}$
- (ii) $\frac{}{s_1 \equiv t_1, \dots, s_n \equiv t_n, \Gamma \vdash A}$ if the system $\{s_1 \equiv t_1, \dots, s_n \equiv t_n\}$ is not solvable.

Proof Slightly differently from the usual presentation (see e.g. [14, 20]), Herbrand’s rules for solving a multiset $S = \{s_1 \equiv t_1, \dots, s_n \equiv t_n\}$ of equations can be described as follows. We consider transformations between pairs $\langle S \mid \sigma \rangle$, where σ is a substitution, which are determined by the following transitions:

$$\begin{aligned} \langle x \equiv x, S \mid \sigma \rangle &\longrightarrow \langle S \mid \sigma \rangle \\ \langle f(r_1, \dots, r_n) \equiv f(r'_1, \dots, r'_n), S \mid \sigma \rangle &\longrightarrow \langle r_1 \equiv r'_1, \dots, r_n \equiv r'_n, S \mid \sigma \rangle \quad (n \geq 0) \\ \left. \begin{aligned} \langle x \equiv t, S \mid \sigma \rangle &\longrightarrow \langle S[t/x] \mid \sigma[t/x] \rangle \\ \langle t \equiv x, S \mid \sigma \rangle &\longrightarrow \langle S[t/x] \mid \sigma[t/x] \rangle \end{aligned} \right\} \text{provided } x \text{ does not occur in } t \end{aligned}$$

Then by comparison with the usual transitions it can easily be seen that σ is an mgu solving S if and only if there is a derivation

$$\langle S \mid \emptyset \rangle \longrightarrow \dots \longrightarrow \langle \emptyset \mid \sigma \rangle. \quad ^5$$

⁵ The difference from the usual procedure is that we “store” equations in solved form as substitutions in the right component of the pair. This avoids the repetition of an unsolved equation $x \equiv t$ in the usual rule $\{x \equiv t\} \cup S \longrightarrow \{x \equiv t\} \cup S[t/x]$, which logically corresponds to contraction.

Now for each step $\langle S \mid \sigma \rangle \longrightarrow \langle S' \mid \sigma' \rangle$ the rule

$$\frac{S', \Gamma \sigma' \vdash A \sigma'}{S, \Gamma \sigma \vdash A \sigma}$$

is a primitive rule in **E** — either $(\equiv \vdash)$, (inj) or $(Repl)$. Therefore from

$$\langle S \mid \emptyset \rangle \longrightarrow \dots \longrightarrow \langle \emptyset \mid \sigma \rangle$$

we obtain a derivation

$$\begin{array}{c} \Gamma \sigma \vdash A \sigma \\ \vdots \\ S, \Gamma \vdash A \end{array}$$

in **E**. Furthermore, S is not solvable if and only if any derivation

$$\langle S \mid \emptyset \rangle \longrightarrow \dots$$

ends at a stage $\langle S' \mid \sigma' \rangle$ such that any equation in S' has the form $x \equiv t$ with x occurring as a proper subterm in t , or the form $f(r_1, \dots, r_n) \equiv g(r_1', \dots, r_m')$ ($m, n \geq 0$) with g distinct from f . Since by (\neq) and (occ) we have $S', \Gamma \sigma' \vdash A \sigma'$ in these cases, we obtain a derivation of $S, \Gamma \vdash A$ in **E**.

Lemma 2 (*Substitution lemma*) $\mathbf{DRE}(\mathcal{D})$ is closed under substitution, i.e., if $\Gamma \vdash A$ is derivable in $\mathbf{DRE}(\mathcal{D})$, then so is $\Gamma \theta \vdash A \theta$ for any substitution θ .

Proof We proceed by induction on the length of derivations and consider the last step of the given derivation of $\Gamma \vdash A$.

Apart from (occ) , rules without premisses are closed under substitution. In the case of (occ) we observe that $x\theta$ and $t\theta$ are not unifiable, if x occurs as a proper subterm of t , so that we can apply Lemma 1 (ii).

In all cases of rules with premisses except $(\mathcal{D}^+)_\omega$ and $(Repl)$, we apply the induction hypothesis to the premiss(es).

In the case

$$\frac{\{\Gamma' \sigma_i, C_i \sigma_i \vdash A \sigma_i\}_i}{\Gamma', a \vdash A} (\mathcal{D}^+)_\omega,$$

where Γ is Γ', a and $\sigma_i = mgu(a, b_i)$ for the i -th clause $b_i \Leftarrow C_i$ in \mathcal{D} if there is a unifier — otherwise the i -th premiss is missing — we proceed as follows. Suppose $\sigma_i' = mgu(a\theta, b_i)$. Then $\theta \sigma_i'$ is a unifier of a and b_i (since variables are standardized apart), so $\theta \sigma_i' = \sigma_i \theta_i$ for some θ_i . By induction hypothesis we obtain

$$\Gamma' \sigma_i \theta_i, C_i \sigma_i \theta_i \vdash A \sigma_i \theta_i, \text{ i.e., } \Gamma' \theta \sigma_i', C_i \theta \sigma_i' \vdash A \theta \sigma_i',$$

from which by $(\mathcal{D}^+)_\omega$

$$\Gamma' \theta, a \theta \vdash A \theta$$

follows.

In the case

$$\frac{\Gamma[t/x] \vdash A[t/x]}{x \equiv t, \Gamma' \vdash A} (\text{Repl}),$$

where Γ is $x \equiv t$, Γ' , we can assume that $x\theta$ and $t\theta$ are unifiable (otherwise the assertion holds by Lemma 1 (ii)). Suppose $\sigma = mgu(x\theta, t\theta)$. Then by induction hypothesis we derive

$$\Gamma'[t/x]\theta\sigma \vdash A[t/x]\theta\sigma,$$

which is the same as

$$\Gamma'\theta\sigma \vdash A\theta\sigma,$$

since x is not in t . By Lemma 1 (i) we obtain

$$x\theta \equiv t\theta, \Gamma'\theta \vdash A\theta.$$

Corollary 1 *The rules $(\text{Repl})'$ and (Thin^{\equiv}) are admissible.*

A rule is called *invertible*, if its inverse is admissible, i.e., if the derivability of its conclusion implies the derivability of each of its premisses. As a generalized inverse of $(\equiv \vdash)$ we consider the rule

$$(\equiv \vdash)^{\vee} \frac{t_1 \equiv t_1, \dots, t_n \equiv t_n, \Gamma \vdash A}{\Gamma \vdash A}$$

Lemma 3 (*Invertibility lemma*) (i) *The rule $(\equiv \vdash)^{\vee}$ is admissible in $\text{DRE}(\mathcal{D})$.*
(ii) *All equality rules with premisses, i.e., $(\equiv \vdash)$, (Repl) and (inj) are invertible in $\text{DRE}(\mathcal{D})$.*

Proof (i): Induction on the length of derivations.

(ii): $(\equiv \vdash)$: Follows from (i).

(Repl) : From $x \equiv t, \Gamma \vdash A$ we obtain

$$t \equiv t, \Gamma[t/x] \vdash A[t/x]$$

by Lemma 2, and

$$\Gamma[t/x] \vdash A[t/x]$$

by $(\equiv \vdash)^{\vee}$.

(inj) : The derivation

$$\frac{}{f(t_1, \dots, t_n) \equiv f(t_1', \dots, t_n') \vdash f(t_1, \dots, t_n) \equiv f(t_1', \dots, t_n')} (I)$$

has to be transformed into

$$\frac{\overline{\vdash f(t_1', \dots, t_n') \equiv f(t_1, \dots, t_n)}^{(\vdash \equiv)}}{t_1 \equiv t_1', \dots, t_n \equiv t_n' \vdash f(t_1, \dots, t_n) \equiv f(t_1', \dots, t_n')} \text{ (Repl)' (n times)}$$

All other rules without premisses are treated in an obvious way. The rules with premisses are permutable with (inj) .

Corollary 2 *The following rules are admissible in DRE(D):*

- (i) $\frac{s_1 \equiv t_1, \dots, s_n \equiv t_n, \Gamma \vdash A}{\Gamma \sigma \vdash A \sigma}$ if σ solves the system of equations $\{s_1 \equiv t_1, \dots, s_n \equiv t_n\}$
- (ii) $(Contr \equiv)$

Proof (i): In Lemma 1 (i) we have shown that the converse of this rule is derivable in **E** by means of $(\equiv \vdash)$, (inj) and $(Repl)$. So we just have to apply the previous lemma.

(ii): If s and t are not unifiable, then $s \equiv t, \Gamma \vdash B$ is derivable by Lemma 1 (ii). If $\sigma = mgu(s, t)$, then σ solves the multiset $\{s \equiv t, s \equiv t\}$ as well as the multiset $\{s \equiv t\}$. Therefore from (i) we obtain $\Gamma \sigma \vdash B \sigma$, from which by Lemma 1 (i) we derive $s \equiv t, \Gamma \vdash B$.

Lemma 4 *(Admissibility of equality cut) The rule*

$$(Cut \equiv) \frac{\Gamma \vdash s \equiv t \quad \Gamma, s \equiv t \vdash A}{\Gamma \vdash A}$$

is admissible in DRE(D).

Proof by induction on the length of the derivation of the left premiss of the cut. We consider the rule applied in the last step of this derivation.

If it is (I) , then the second premiss is already the conclusion.

If it is $(\vdash \equiv)$, then we apply $(\equiv \vdash)^\vee$ (which by Lemma 3 (i) is admissible) to the right premiss, deleting $t \equiv t$ in its antecedens.

If it is any other rule without premiss, then the conclusion of the cut is an application of this rule as well.

If it is $(Repl)$ or $(D\vdash)_\omega$, then we apply Lemma 2 to the right premiss of the cut and apply the induction hypothesis.

In all other cases we can move the cut one step up and apply the induction hypothesis. (Note that $(\vdash D)$ cannot have an equality as its conclusion.)

4 The Completion of \mathcal{D}

Unlike clauses, which define particular instances of a predicate p , the completed definition of p in Clark's [3] sense is uniform in the arguments of p . In our Gentzen-style framework we capture this idea as follows. Suppose the clauses for p in \mathcal{D} are

$$\begin{aligned} p(t_{11}, \dots, t_{1m}) &\Leftarrow C_1 \\ &\vdots \\ p(t_{n1}, \dots, t_{nm}) &\Leftarrow C_n. \end{aligned}$$

Then we define the following Right- and Left-introduction rules for p :

$$(t-p) \frac{\Gamma_1 \vdash t_1 \equiv t_{i_1} \sigma \quad \dots \quad \Gamma_m \vdash t_m \equiv t_{i_m} \sigma \quad \Gamma \vdash C_i \sigma}{\Gamma_1, \dots, \Gamma_m, \Gamma \vdash p(t_1, \dots, t_m)} \quad 1 \leq i \leq n$$

$$(p\vdash) \frac{\{\Gamma, t_1 \equiv t_{i_1}, \dots, t_m \equiv t_{i_m}, C_i \vdash A : 1 \leq i \leq n\}}{\Gamma, p(t_1, \dots, t_m) \vdash A} \quad \begin{array}{l} \text{provided the variables in} \\ t_{ij}, C_i \text{ are not free in } t_i, \Gamma, A \end{array}$$

If p is not defined (i.e., $n = 0$), then there is no $(t-p)$, and $(p\vdash)$ is

$$\overline{\Gamma, p(t_1, \dots, t_m) \vdash A}$$

By $\text{COMP}(\mathcal{D})$ we denote $\text{DRE}(\mathcal{D})$ with $(\vdash\mathcal{D})$ and $(\mathcal{D}\vdash)_\omega$ being replaced with $(t-p)$ and $(p\vdash)$ for all predicates p occurring in \mathcal{D} .

Now let \mathcal{C}_p be the following formula, which according to Clark [3] expresses the completed definition of the predicate p :

$$(\forall x_1 \dots x_m)(p(x_1, \dots, x_m) \leftrightarrow ((\exists \bar{y}_1)(x_1 = t_{11} \wedge \dots \wedge x_m = t_{1m} \wedge C_1) \vee \dots \vee (\exists \bar{y}_n)(x_1 = t_{n1} \wedge \dots \wedge x_m = t_{nm} \wedge C_n)))$$

where \bar{y}_i contains the variables free in the i -th clause for p . If $n = 0$, then \mathcal{C}_p is

$$(\forall x_1 \dots x_m)(p(x_1, \dots, x_m) \leftrightarrow \perp), \text{ i.e., } (\forall x_1 \dots x_m) \neg p(x_1, \dots, x_m).$$

Then we can easily derive \mathcal{C}_p in $\text{COMP}(\mathcal{D})$. Conversely, in the presence of cut, we can derive $(t-p)$ and $(p\vdash)$ using \mathcal{C}_p as an axiom.

It is not difficult to check that all results proved in the previous section for $\text{DRE}(\mathcal{D})$ also hold for $\text{COMP}(\mathcal{D})$. In fact, the proofs become simpler since it is immediately obvious that $(p\vdash)$ is closed under substitution and can be permuted with (Repl) , because no unifiers are involved. However, we do not have to spell out proofs since these results are a consequence of what follows. We only need Lemma 1, which, as an assertion about *derivability* in \mathbf{E} , also holds for $\text{COMP}(\mathcal{D})$.

5 Definitional Reflection and the Completion

The main result of this paper is the equivalence of $\text{DRE}(\mathcal{D})$ and $\text{COMP}(\mathcal{D})$. It is almost a corollary of what we have proved in Section 3.

Theorem 1 *Each primitive rule of $\text{COMP}(\mathcal{D})$ is admissible in $\text{DRE}(\mathcal{D})$, and each primitive rule of $\text{DRE}(\mathcal{D})$ is derivable in $\text{COMP}(\mathcal{D})$.*

Proof We have to consider $(\vdash\mathcal{D})$, $(\mathcal{D}\vdash)_\omega$, $(t-p)$ and $(p\vdash)$. The rest is identical in the two systems.

(i) $(\vdash p)$ is admissible in $\mathbf{DRE}(\mathcal{D})$.

$$\frac{\frac{\Gamma \vdash C_i \sigma}{\Gamma \vdash p(t_{i1}\sigma, \dots, t_{im}\sigma)} (\vdash \mathcal{D})}{\frac{\Gamma_1 \vdash t_1 \equiv t_{i1}\sigma \dots \Gamma_m \vdash t_m \equiv t_{im}\sigma \quad t_1 \equiv t_{i1}\sigma, \dots, t_m \equiv t_{im}\sigma, \Gamma \vdash p(t_1, \dots, t_m)}{\Gamma_1, \dots, \Gamma_m, \Gamma \vdash p(t_1, \dots, t_m)} (\text{Repl})'} (\text{Cut} \equiv)$$

(ii) $(p \vdash)$ is admissible in $\mathbf{DRE}(\mathcal{D})$.

If $\{t_1 \equiv t_{i1}, \dots, t_m \equiv t_{im}\}$ is not solvable, then $p(t_1, \dots, t_m)$ and $p(t_{i1}, \dots, t_{im})$ are not unifiable. Otherwise, let σ_i be such a unifier. Then we obtain

$$\frac{\left\{ \frac{\Gamma, t_1 \equiv t_{i1}, \dots, t_m \equiv t_{im}, C_i \vdash A}{\Gamma \sigma_i, C_i \sigma_i \vdash A \sigma_i} \text{Corollary 2 (i)} \right\}_i (\mathcal{D} \vdash)_\omega}{\Gamma, p(t_1, \dots, t_m) \vdash A}$$

(iii) $(\vdash \mathcal{D})$ is derivable in $\mathbf{COMP}(\mathcal{D})$.

Let a be $p(t_{i1}, \dots, t_{im})$.

$$\frac{\frac{\vdash t_{i1}\sigma \equiv t_{i1}\sigma (\vdash \equiv) \dots \vdash t_{im}\sigma \equiv t_{im}\sigma (\vdash \equiv)}{\Gamma \vdash a \sigma} (\vdash p)}{\Gamma \vdash C_i \sigma} (\vdash p)$$

(iv) $(\mathcal{D} \vdash)_\omega$ is derivable in $\mathbf{COMP}(\mathcal{D})$.

Let a be $p(t_1, \dots, t_m)$. If $\sigma_i = \text{mgu}(p(t_1, \dots, t_m), p(t_{i1}, \dots, t_{im}))$, then σ_i solves the equation system $\{t_1 \equiv t_{i1}, \dots, t_m \equiv t_{im}\}$. Therefore we have

$$\frac{\left\{ \frac{\Gamma \sigma_i, C_i \sigma_i \vdash A \sigma_i}{\Gamma, t_1 \equiv t_{i1}, \dots, t_m \equiv t_{im}, C_i \vdash A} \text{Lemma 1} \right\}_i (p \vdash)}{\Gamma, a \vdash A}$$

6 Defined Equality

Instead of formulating a separate theory of free equality one may treat equality directly in the framework of $\mathbf{DR}(\mathcal{D})$. As has been proposed independently by Eriksson [5] and Girard [8], we can *define* equality by extending a definition \mathcal{D} with the single clause

$$z \equiv z \Leftarrow \top .$$

By “ \mathcal{D}^\equiv ” we denote \mathcal{D} enlarged with this equality clause. Then we can show that $\mathbf{DR}(\mathcal{D}^\equiv)$ is as powerful as $\mathbf{DRE}(\mathcal{D})$.

Lemma 5 (i) *The rule (inj) is admissible in $\mathbf{DR}(\mathcal{D}^\equiv)$, while any other rule of $\mathbf{DRE}(\mathcal{D})$ is derivable in $\mathbf{DR}(\mathcal{D}^\equiv)$.*

(ii) *Every rule of $\mathbf{DR}(\mathcal{D}^\equiv)$ is derivable in $\mathbf{DRE}(\mathcal{D})$.*

Proof (i) For $(\vdash \equiv)$ and $(\equiv \vdash)$ this is obvious (we have to use $(\vdash \top)$ and $(\top \vdash)$). For (Repl) we use $(\top \vdash)$ and the fact that $\{t/x, t/z\}$ is an mgu of $x \equiv t$ and $z \equiv z$. For

(\neq) and (*occ*) we use that neither $f(t_1, \dots, t_n) \equiv g(t'_1, \dots, t'_n)$ (f and g distinct) nor $x \equiv t$ (x proper subterm of t) unifies with $z \equiv z$. For (*inj*) we argue as follows. If $f(t_1, \dots, t_n)$ and $f(t'_1, \dots, t'_n)$ are not unifiable, then the conclusion of (*inj*) holds by $(\mathcal{D}^{\equiv\vdash})_\omega$ anyway. Otherwise, for a unifier σ , from $t_1 \equiv t'_1, \dots, t_n \equiv t'_n, \Gamma \vdash A$ we obtain $t_1 \sigma \equiv t'_1 \sigma, \dots, t_n \sigma \equiv t'_n \sigma, \Gamma \sigma \vdash A \sigma$, since $\mathbf{DR}(\mathcal{D}^{\equiv})$ is closed under substitution. Since the rule $(\equiv\vdash)^\vee$ can easily be shown admissible in $\mathbf{DR}(\mathcal{D}^{\equiv})$, we obtain $\Gamma \sigma \vdash A \sigma$, from which by $(\top\vdash)$ and $(\mathcal{D}^{\equiv\vdash})_\omega$ the desired result $f(t_1, \dots, t_n) \equiv f(t'_1, \dots, t'_n), \Gamma \vdash A$ follows.

(ii) We use $(\top\vdash)$ and Lemma 1.

Remark Instead of using the clause $z \equiv z \leftarrow \top$ for equality, one can of course regard the following rules, which are obtained from this clause, as an equality theory of its own (see [8]):

$$\frac{}{\vdash t \equiv t} \quad \frac{\Gamma \sigma \vdash A \sigma}{s \equiv t, \Gamma \vdash A} \text{ if } \sigma = mgu(s, t) \quad \frac{}{s \equiv t, \Gamma \vdash A} \text{ if } s \text{ and } t \text{ are not unifiable}$$

As a consequence of Lemma 5 we obtain the following theorem.

Theorem 2 *COMP(\mathcal{D}) and $\mathbf{DR}(\mathcal{D}^{\equiv})$ are equivalent in the sense that each rule of one system is admissible in the other one.*

Thus we obtain a computational interpretation of $\mathbf{COMP}(\mathcal{D})$ in so far as $\mathbf{DR}(\mathcal{D}^{\equiv})$ has a computational interpretation. However, as remarked in [18], this computational interpretation of $\mathbf{DR}(\mathcal{D})$ (and therefore $\mathbf{DR}(\mathcal{D}^{\equiv})$, which is a special case thereof) does not go very far, since variables in a sequent $\Gamma \vdash A$, which is to be evaluated by $(\mathcal{D}^{\vdash})_\omega$, are understood universally rather than existentially. It is not an easy task at all to combine this feature with the idea of computing answer substitutions, since in $\mathbf{DR}(\mathcal{D})$ we do not have lifting (permutability of substitution and computation) in the usual sense (see [6] for a presentation of some difficulties). This is due to the fact that with respect to computation $(\vdash\mathcal{D})$ and $(\mathcal{D}^{\vdash})_\omega$ are not completely dual to each other. Loosely speaking, the left side of the turnstile corresponds to negative goals and the right side to positive goals in traditional logic programming. Actually, in some sense $(\mathcal{D}^{\vdash})_\omega$ corresponds to the negation by failure rule, whereas $(\vdash\mathcal{D})$ corresponds to ordinary resolution. This relationship has to be elaborated in subsequent work.

7 Final Remarks

Substructural Logics

Logics with restricted structural rules (substructural logics) are particularly interesting in combination with definitional reflection. The logic $\mathbf{DR}(\mathcal{D})$ does not enjoy eliminability of cut in general whereas the contraction-free variant does (see [17]). It can easily be checked that all results and proofs in Sections 3 to 6 are independent of whether in the underlying logic $\mathbf{DR}(\mathcal{D})$ contraction or thinning or both of them are lacking. This is because in the equality extension $\mathbf{DRE}(\mathcal{D})$ we have restricted the rules of thinning and contraction to formulas which are not equalities. Instead we have shown that for equalities both thinning and contraction are admissible independently of their availability for arbitrary formulas. Furthermore, we have formulated the rules

for \top and \wedge in the multiplicative way in Girard's [7] sense (his notation would be $\mathbf{1}$ and \otimes , respectively). The addition of additive truth and additive conjunction does not cause any change in the proofs.

Our proof of cut elimination for the contraction-free case [17] can easily be extended to the system with equality. We just apply the reductions given in the cut elimination proof and eliminate equality cuts according to Lemma 4 whenever we encounter such cuts. We only have to check that the number of applications of $(\mathcal{D}^\dagger)_\omega$, which is one of the induction parameters of the cut elimination proof, is not increased by the elimination of equality cuts, which is easily done.

Multiple Succedent Logics

Allowing for more than one formula in the succedent of a sequent does not change the results obtained. This is because both the equality rules and $(\mathcal{D}^\dagger)_\omega$ (which produce substitutions even in the parametric parts of an inference rule) essentially operate on the left side of a sequent, so that all the basic lemmas of Section 3 remain valid. Therefore neither for classical logic (system $\mathbf{DRE}(\mathcal{D})$ with multiple succedents) nor for substructural logics (multiple succedent with additional multiplicative constants for falsity and disjunction) we have to deal with cases which crucially differ from what we have already considered.

Restricted Initial Sequents

Kreuger [12] has argued that a system, in which the rule

$$(I) \frac{}{a \vdash a}$$

is restricted to specific atoms a , is more appropriate for definitional reflection than a system with unrestricted (I) . Actually, cut elimination holds for certain ways of restricting (I) , even when contraction is present (see [19]). In any case the rule

$$(I^\equiv) \frac{}{r \equiv s \vdash r \equiv s}$$

is derivable (due to $(\vdash \equiv)$ and the admissibility of $(\mathit{Repl})'$).

Nothing of what we have proved in the previous sections hinges upon having (I^\equiv) as a primitive rather than derived rule of inference. Similar to the contraction-free case, the cut elimination proof for the system with restricted (I) can be carried over to the system with equality.

Jäger & Stärk's Approach

In their proof-theoretic interpretation of PROLOG's negation-as-failure rule, Jäger & Stärk [10] use a system which can be described as a multiple succedent variant of our $\mathbf{COMP}(\mathcal{D})$ with negation as a primitive constant (rather than implication and absurdity), and an axiomatic rather than a rule-based theory of free equality (which, therefore, is not cut-free). Actually, Stärk's [21] system with "quantifier-free program rules" comes closest to $\mathbf{COMP}(\mathcal{D})$. Based on a four-valued semantics, according to which a sequent expresses that either the antecedent is false or the succedent is true, Jäger & Stärk reject (I) . By a method different from ours [17], they prove that cuts

with formulas, which are not equalities, can be eliminated (see [10]). This method can easily be adapted to the contraction-free case and carried over to our system. The difference in principle between their approach and ours concerns research goals. Jäger & Stärk basically want to understand SLDNF resolution by proof-theoretic means, whereas we are interested in definitional reflection as a fundamental proof-theoretic principle in systems like $\mathbf{DR}(\mathcal{D})$.

Conclusion

The equivalence between $\mathbf{DR}(\mathcal{D})$ and the completion of a definition in the presence of equality indicates the declarative strength of $(\mathcal{D}\vdash)_\omega$. However, computationally, the system for definitional reflection based on $(\mathcal{D}\vdash)$ is preferable to the system $\mathbf{DR}(\mathcal{D})$ (which is based on $(\mathcal{D}\vdash)_\omega$), since it permits the computation of bindings to variables independent of their (positive or negative) position in hypothetical queries. This feature is exploited in the programming language GCLA [1, 2].

References

1. Aronsson, M., *GCLA: The Design, Use, and Implementation of a Program Development System*. Ph.D. thesis, University of Stockholm 1993.
2. Aronsson, M., Eriksson, L.-H., Gäredal, A., Hallnäs, L. & Olin, P. The programming language GCLA: A definitional approach to logic programming. *New Generation Computing*, 4 (1990), 381–404.
3. Clark, K. L. Negation as failure. In: Gallaire, H. & Minker, J. (Eds.), *Logic and Data Bases*, New York 1978, 293–322.
4. Coquand, T. Pattern matching with dependent types. In: *Workshop on Logical Frameworks, Båstad 1992, Proceedings*, available by ftp from ftp.cs.chalmers.se as /pub/cs/reports/baastad.92/procSS.
5. Eriksson, L.-H. A finitary version of the calculus of partial inductive definitions. In: Eriksson, L.-H., Hallnäs, L. & Schroeder-Heister, P. (Eds.), *Extensions of Logic Programming. Second International Workshop, ELP-91, Stockholm, January 1991, Proceedings*. Springer LNCS, Vol. 596, Berlin 1992, 89–134.
6. Eriksson, L.-H. *Finitary Partial Inductive Definitions and General Logic*. Ph.D. thesis, Royal Institute of Technology, Stockholm 1993.
7. Girard, J.-Y. Linear logic. *Theoretical Computer Science*, 50 (1987), 1–102.
8. Girard, J.-Y. A fixpoint theorem for linear logic. In: P. Lincoln (Ed.), *Linear Logic Mailing List*, linear@cs.stanford.edu, 5 February 1992 (Reply: *ibid.*, 19 February 1992).
9. Hallnäs, L. & Schroeder-Heister, P. A proof-theoretic approach to logic programming. I. Clauses as rules. *Journal of Logic and Computation*, 1 (1990), 261–283; II. Programs as definitions, *ibid.* 1 (1991), 635–660. Originally published as SICS Research Report 88005, 1988.

10. Jäger, G. & Stärk, R. F. A proof-theoretic framework for logic programming. In: S. Buss (Ed.), *Handbook of Proof Theory* (forthcoming).
11. Kanger, S. A simplified proof method for elementary logic. In: Braffort, P. & Hirschberg, D. (Eds.), *Computer Programming and Formal Systems*, Amsterdam 1963, 87–94.
12. Kreuger, P. Axioms in definitional calculi. This volume.
13. Lifshits, V.A. Normal form for deductions in predicate calculus with equality and functional symbols. In: Slisenko, A.O. (Ed.), *Studies in Constructive Mathematics and Mathematical Logic I*, New York 1969, 21–23.
14. Martelli, A. & Montanari, U. An efficient unification algorithm. *ACM Transactions on Programming Languages and Systems*, 4 (1982), 259–282.
15. Richter, M.M. *Logikkalküle*. Teubner, Stuttgart 1978.
16. Sahlin, D., Franzén, T. & Haridi, S. An intuitionistic predicate logic theorem prover. *Journal of Logic and Computation* 2 (1992), 619–656.
17. Schroeder-Heister, P. Cut-elimination in logics with definitional reflection. In: D. Pearce & H. Wansing (Eds.), *Nonclassical Logics and Information Processing. International Workshop, Berlin 1990, Proceedings*. Springer LNCS, Vol. 619, Berlin 1992, 146–171.
18. Schroeder-Heister, P. Rules of definitional reflection. In: *8th Annual IEEE Symposium on Logic in Computer Science (Montreal 1993)*. IEEE Computer Society Press, Los Alamitos 1993, 222–232.
19. Schroeder-Heister, P. Cut elimination for logics with definitional reflection and restricted initial sequents. Manuscript, available by ftp as /pub/LS/resini from gopher.informatik.uni-tuebingen.de.
20. Snyder, W. *A Proof Theory for General Unification*. Birkhäuser, Basel 1991.
21. Stärk, R. F. Cut-property and negation as failure. Technical report, Institut für Informatik und angewandte Mathematik, Universität Bern, 1992 (available by ftp as /pub/staerk/cut from ftp.cis.uni-muenchen.de).