Logica Universalis 1 (2), 2007, in press

Generalized Definitional Reflection and the Inversion Principle

Peter Schroeder-Heister
Wilhelm-Schickard-Institut
Universität Tübingen
Sand 13
72076 Tübingen
Germany

May 30, 2007

Abstract

The term *inversion principle* goes back to Lorenzen who coined it in the early 1950s. It was later used by Prawitz and others to describe the symmetric relationship between introduction and elimination inferences in natural deduction, sometimes also called *harmony*. In dealing with the invertibility of rules of an arbitrary atomic production system, Lorenzen's inversion principle has a much wider range than Prawitz's adaptation to natural deduction,. It is closely related to *definitional reflection*, which is a principle for reasoning on the basis of rule-based atomic definitions, proposed by Hallnäs and Schroeder-Heister. After presenting definitional reflection and the inversion principle, it is shown that the inversion principle can be formally derived from definitional reflection, when the latter is viewed as a principle to establish *admissibility*. Furthermore, the relationship between definitional reflection and the inversion principle is investigated on the background of a *universalization principle*, called the ω-principle, which allows one to pass from the set of all defined substitution instances of a sequent to the sequent itself.

1. Definitional Reflection

In proof-theoretic semantics, inversion principles are central to unifying approaches to logic and logical constants. Prawitz [13] describes the uniform relationship between introduction and elimination inferences in natural deduction by such a principle, which justifies the elimination inferences as a sort of 'consequence' of the introduction inferences, in accordance with certain remarks made by Gentzen [3]. He explicitly adopts the term 'inversion principle' from Lorenzen, who coined it in the early 1950s [10, 11]. Prawitz's later definition of validity for derivations in natural deduction style can be understood as a further elaboration of inversion, making the *justification* of inferences explicit in the form of reduction procedures associated with noncanonical derivations.[1] The basic idea behind this inversion principle is that if certain defining ('canonical') inferences for A, say

$$\frac{B_1}{A} \quad \cdots \quad \frac{B_n}{A}$$

are given, then an inference with A as a premiss, say

$$\frac{A}{C}$$

is justified if each defining condition B_i of A entails the conclusion C, in other words: if C can be derived from each B_i, then C can be derived from A. Putting this principle into an inference schema yields

$$\frac{A \qquad \begin{matrix} B_1 \\ C \end{matrix} \quad \cdots \quad \begin{matrix} B_n \\ C \end{matrix}}{C}$$

This pattern, which corresponds to standard $\vee$-elimination, leads to a uniform schema for elimination rules in natural deduction[2].

However, the idea of inversion is not confined to logic and logical constants. In the form of *definitional reflection* as proposed by Hallnäs and Schroeder-Heister it becomes is a general principle of definitional reasoning.[3] Suppose for simplicity that

[1] See, e.g., [14, 16], and, for a critical overview, [23].

[2] See [18]. For the relationship between the uniform schema and generalized elimination rules see [22]. A discussion of Lorenzen's inversion principle in relation to the proof-theoretic justification of logical laws can be found in [12] and [24, 25].

[3] As a schema to universally characterize logical constants, a principle like $(\mathcal{D}\vdash)$ was proposed by Schroeder-Heister[18], leading to generalized elimination rules and correcting an error made by Prawitz [15] in a paper with a similar target. The idea and the formulation of $(\mathcal{D}\vdash)$ as a general principle for definitional reasoning independent of the application to logical constants, and the idea to use it for a nonmonotonic extension of logic programming is due to Hallnäs [1, 4, 5]. He also proposed the terminology 'definitional closure / definitional reflection' (see below). The further proof-theoretic elaboration of definitional reflection in the presence of free variables, both in the context of logic programming and in general logic, was carried out by Hallnäs and Schroeder-Heister [6, 19, 20, 21].

capital letters are propositional constants. Let expressions of the form $A_1, \ldots, A_n \to A$, also written as $\Gamma \to A$, be called *rules*. Suppose a certain finite set of rules

$$\mathcal{D} \left\{ \begin{array}{lcl} A_1 & \leftarrow & \Delta_1 \\ & \vdots & \\ A_m & \leftarrow & \Delta_m \end{array} \right.$$

is considered to be a *definition*.[4] Suppose that

$$\left\{ \begin{array}{lcl} A & \leftarrow & \Gamma_1 \\ & \vdots & \\ A & \leftarrow & \Gamma_n \end{array} \right.$$

are exactly those rules in $\mathcal{D}$, whose head is A. They are called the *defining rules* of A (with respect to $\mathcal{D}$). The set of *defining conditions* $\{\Gamma_1, \ldots, \Gamma_n\}$ of A is denoted by $\mathcal{D}(A)$. As a limiting case, a defining condition may be empty, in which case the rule is written as

$$A \leftarrow$$

(in logic programming terminology, A is a *fact*). Note that this situation must be distinguished from $\mathcal{D}(A) = \emptyset$, i.e., from there being no defining condition of A at all. Then the principle of *definitional reflection* says that everything that is entailed by each defining condition of A is entailed by A itself (in any context Δ). Formulated as an inference principle in the sequent calculus this reads as

$$\frac{\Gamma_1, \Delta \vdash C \quad \ldots \quad \Gamma_n, \Delta \vdash C}{A, \Delta \vdash C} \; (\mathcal{D} \vdash)$$

in short

$$\frac{\{\Gamma, \Delta \vdash C : \Gamma \in \mathcal{D}(A)\}}{A, \Delta \vdash C} \; (\mathcal{D} \vdash)$$

This principle complements the principle of *definitional closure*

$$\frac{\Delta \vdash A_{i1} \quad \ldots \quad \Delta \vdash A_{ik_i}}{\Delta \vdash A} \; (\vdash \mathcal{D})$$

in short

$$\frac{\Delta \vdash \Gamma_i}{\Delta \vdash A} \; (\vdash \mathcal{D})$$

[4] Following the practice in logic programming I often write rules with the head in leading position, but only, if they are definitional, i.e., primitive ('program rules' in the terminology of logic programming).

if $\Delta \vdash \Gamma_i$ is read in the obvious way as expressing a list of sequents[5]. Definitional closure expresses the *application* of definitional rules in the obvious way.

Concerning terminology, I speak of *inferences* or *inference principles* to distinguish rules in sequent systems from rules for atoms of the form $\Gamma \to A$, for which I want to reserve the term *rule*[6].

The principles of definitional closure and definitional reflection can also be formulated in natural deduction style as

$$\frac{\Gamma_i}{A} \text{ (definitional closure)}$$

and

$$\frac{A \quad \left\{ \begin{array}{c} \Gamma \\ C \end{array} : \Gamma \in \mathcal{D}(A) \right\}}{C} \text{ (definitional reflection).}$$

Formulated for propositional constants as atoms, these principles are, of course, not very powerful. However, they illustrate the underlying idea. They are added as inference principle for atoms to a logical system L, yielding an extended system $L(\mathcal{D})$, called the *definitional logic based on* $\mathcal{D}$. Since systems of atomic rules can be identified with inductive definitions, $L(\mathcal{D})$ may be viewed as a logic extended with an inductive definition. In $L(\mathcal{D})$, the symmetry pattern of logical principles (sometimes called *harmony*) is extended by a corresponding pattern of principles for definitional reasoning: The rules of a definition give rise to sequent style right introduction and left introduction inferences for atoms (or natural deduction style introduction and elimination inferences for atoms). Depending on what one is interested in, the underlying system L may be a full-fledged logical system such as first-order logic, or a more parsimonious system containing some standard structural principles such as identity, thinning and contraction

$$\frac{}{A \vdash A} \text{ (Id)} \qquad \frac{\Gamma \vdash A}{\Gamma, \Delta \vdash A} \text{ (Thin)} \qquad \frac{\Gamma, B, B \vdash A}{\Gamma, B \vdash A} \text{ (Contr)}$$

[5] Not to be confounded with a multiple succedent sequent, where Γ in $\Delta \vdash \Gamma$ is understood disjunctively. Sequents of this kind are not considered in this paper.

[6] For the latter I could, of course, have chosen a different term such as *clause* or *production*. In fact, *production* would not be bad (unlike *clause*, which suggests a disjunctive reading common in resolution theory). However, in the present context, I prefer to stick to Lorenzen's terminology, who speaks of *rules*. As will be seen in section 3, speaking of *inference principles* in connection with sequents fits also well with Lorenzen's terminology, if sequents are read as expressing admissibility statements, so that rules for sequents are *principles for establishing admissibility*.

or even a weaker substructural logic. I do not investigate here which impact different choices of the structural and logical properties of the basic system L have for features of the resulting definitional logic $L(\mathcal{D})$. For simplicity, I assume that the antecedent Γ of a sequent $\Gamma \vdash A$ is a *multiset*, and that the three structural principles (Id), (Thin) and (Contr) just mentioned are available. However, for the points made in this paper, nothing really hinges on that point.

As the logical inferences themselves follow the general symmetry pattern, they can be interpreted within the definitional schema. Using for the time being a language with individual variables $p, q, r, \ldots$, term forming operations $\wedge$, $\vee$ and $\supset$ (binary) and $\bot$ (nullary), and a single unary predicate T ('truth'), consider the following definition:

$$\mathcal{D}_1 \begin{cases} T(p \wedge q) & \leftarrow \quad T(p), T(q) \\ T(p \vee q) & \leftarrow \quad T(p) \\ T(p \vee q) & \leftarrow \quad T(q) \\ T(p \supset q) & \leftarrow \quad (T(p) \Rightarrow T(q)) \\ \text{[no defining rule for } T(\bot)] \end{cases}$$

Here I temporarily assume not only the first-order structure of atomic formulas to be given (it will be formally introduced and discussed in section 3), but I also suppose that in the underlying logic L some sort of implication ' $\Rightarrow$ ' is available. Given an appropriate logic L with such an implication, from the system $L(\mathcal{D}_1)$ the standard inference principles of intuitionistic propositional logic can be extracted. This cannot just be discarded as a duplication of constants between L and $\mathcal{D}_1$, arguing that one already presupposes what one wants to define, as at least for disjunction and absurdity no corresponding constant in L need to be considered (unlike defined implication '$\supset$', which corresponds to primitive implication ' $\Rightarrow$ ', and defined conjunction '$\wedge$', which corresponds to the comma)[7].

I do not want to further discuss these foundational issues here, but just mention the following points:

1. There is no restriction on the rules in a definition $\mathcal{D}$ in the sense that the premisses of a definitional rule must only contain atoms already defined otherwise. As in logic programming, definitional rules are completely arbitrary, thus permitting, e.g., circular definitions. This opens up new possibilities of dealing with non-wellfounded phenomena such as paradoxes[8].

[7] Obviously, absurdity '$\bot$' is a special case, presupposing a certain interpretation of limiting cases of ($\mathcal{D}\vdash$) if no defining rule for an atom is given.

[8] See [22]. Sometimes, in logic programming, certain well-foundedness properties for definitions are discussed, such as the properties of a program being *hierarchical* or *stratified*. However, this is not considered to be a requirement for a program to make sense, but as something, which, when it holds, influences the global behaviour of the program (such as the functioning of *negation as failure*, see,

2. Implications in the bodies of rules gives the logic of definitional reflection special power which goes beyond what is normally considered (e.g. in the theory of monotone inductive definitions or in definite clause logic programming). When such an implication is available, certain definitions are partial in the sense that, e.g., the eliminability of cuts is lost, i.e., the system lacks certain global features, though locally it behaves 'nicely'[9].

In the following I shall only consider rules of the form $A_1, \dots, A_n \rightarrow A$ for atoms A (which may contain individual variables, see section 3), i.e., rules without implications in bodies. This means that logical implication cannot be expressed in the systems considered here. This corresponds to the usual way of dealing with atomic systems and inductive definitions and is sufficient for the points I want to make.

2. Definitional reflection as an admissibility principle

Definitional reflection has been developed in the context of partial inductive definitions (see [4]), where it is considered to be a fundamental principle of reasoning which is dual to the (more common) principle of definitional closure. In the present context, however, where I want to compare it with the inversion principle proposed by Lorenzen [10, 11] in the early 1950s, a narrower interpretation is appropriate: the *admissibility interpretation*. Given a calculus K as a 'definition', a rule R is admissible in K if adding R to the primitive rules of K does not extend the set of derivable formulas, i.e.,

$$\text{if } \vdash_{K+R} A, \text{ then } \vdash_K A.$$

(The term *admissible* was coined by Lorenzen in this sense.) In order to show that R is admissible, one has to give an *elimination procedure* (again a term proposed by Lorenzen), demonstrating that every application of R in a derivation within $K + R$ can be eliminated, yielding a derivation in K. I now interpret the sequent $\Gamma \vdash A$ as expressing that the rule $\Gamma \rightarrow A$ is admissible (with respect to a calculus given by a set $\mathcal{D}$ of definitional rules). Then both the inference schemata of definitional closure and of definitional reflection become admissibility principles, stating that, if their premisses are valid admissibility statements for $\mathcal{D}$, then so is their conclusion. This is obvious for definitional closure

$$\frac{\Gamma \vdash B_1 \quad \dots \quad \Gamma \vdash B_n}{\Gamma \vdash A}$$

e.g., [8]).

[9] See [20], [22].

where $B_1, \ldots, B_n \to A$ is a primitive rule of $\mathcal{D}$: In order to eliminate an application of $\Gamma \to A$, one may pass from Γ to $B_1, \ldots, B_n$ using the admissible rules $\Gamma \to B_1$, $\ldots$ $\Gamma \to B_n$, and then apply the primitive (= definitional) rule $B_1, \ldots, B_n \to A$. In the case of definitional reflection

$$\frac{\{\Gamma, \Delta \vdash C : \Gamma \in \mathcal{D}(A)\}}{A, \Delta \vdash C}\ (\mathcal{D}\vdash\ )$$

the argument runs as follows: Consider an application of the rule $A, \Delta \to C$. Then its premiss A is derived by using one of the primitive rules $\Gamma \to A$ (with $\Gamma \in \mathcal{D}(A)$) in the last step. Hence there are derivations of the elements of Γ in the previous steps. There are also derivations of the elements of Δ (as premisses of the considered application of $A, \Delta \to C$). Using the admissibility of $\Gamma, \Delta \to C$ one may pass to C as required.

If definitional reflection is understood as an admissibility principle, this is its justification. It is exactly the kind of reasoning Lorenzen uses to justify his inversion principle, if the latter is restricted to the propositional case (see section 4). Abstracting from features irrelevant in our context, in the propositional case Lorenzen's inversion principle is the same as definitional reflection.

3. Definitional reflection in the presence of variables

The situation becomes more complicated when first-order atoms with variables are considered. This is indispensable not only for logic programming, but for definitional reasoning in general. The consideration of the propositional case in the previous sections just served as an outline of the general principle.

I now suppose that atoms of a given signature are built up using predicate symbols (denoted by $P, Q, R, \ldots$), which are applied to terms constructed from constants $(a, b, c, \ldots)$, variables $(x, y, z, \ldots)$ and perhaps function symbols $(f, g, h, \ldots)$. Nullary predicate symbols, which correspond to propositional variables, are allowed as a limiting case. Atoms are denoted by $A, B, C, \ldots$, terms by $t_1, t_2, \ldots$. The set of variables occurring in a term, formula, list of formulas or rule E is denoted by $var(E)$.

This setting is still less general than the situation considered by Lorenzen [11] who does not deal with a term/formula structure but just with a word structure over an arbitrary alphabet. However, it is sufficiently general for most purposes, and in particular, for the foundational issues discussed in this paper.

In the following, a substitution σ is considered to be a partial function from variables to terms whose domain $dom(\sigma)$ is finite (letters for substitutions: $\theta, \rho, \sigma, \tau, \ldots$). Thus substitutions can be identified with finite sets of pairs of variables and terms, written as $[{}^{x_1}/_{t_1}, \ldots, {}^{x_n}/_{t_n}]$. Expressions like $\sigma \cup \tau$ and $\sigma\tau$ are then understood in the usual way, with $\sigma \cup \tau$ only being defined if σ and τ agree on $dom(\sigma) \cap dom(\tau)$ (especially if $dom(\sigma) \cap dom(\tau) = \emptyset$), and with $dom(\sigma \cup \tau) = dom(\sigma) \cup dom(\tau) = dom(\sigma\tau)$ in that case. As usual, a one-one substitution σ mapping variables to variables is called a *renaming substitution* for E, if $var(E) \subseteq dom(\sigma)$. If $\overline{x}$ and $\overline{y}$ are finite sets of variables of the same cardinality, then $[^{\overline{x}}/_{\overline{y}}]$ denotes the substitution, which renames the variables of $\overline{x}$ by the corresponding variables of $\overline{y}$, where for the sake of uniqueness, a standard ordering of all variables is assumed[10]. A rule R' is called a *variant* of a rule R, if $R' = R\sigma$ for a renaming substitution σ for R. A definition $\mathcal{D}'$ is called a variant of a definition $\mathcal{D}$, if $\mathcal{D}'$ results from $\mathcal{D}$ by replacing rules in $\mathcal{D}$ by variants thereof. If σ is a unifier of A and B, it is always assumed that σ affects no variables outside A and B, i.e., $dom(\sigma) \subseteq var(A) \cup var(B)$.

Let again a definition (database, program, calculus) $\mathcal{D}$ of rules

$$\left\{ \begin{array}{lcl} A_1 & \leftarrow & \Gamma_1 \\ & \vdots & \\ A_n & \leftarrow & \Gamma_n \end{array} \right.$$

be given, where the atoms $A_1, \ldots, A_n$ and the atoms in $\Gamma_1, \ldots, \Gamma_n$ are now formulas in the sense just defined. Then the principles of definitional closure and definitional reflection can be formulated as follows:

$$\frac{\Gamma \vdash B_1\sigma \quad \ldots \quad \Gamma \vdash B_m\sigma}{\Gamma \vdash A\sigma}\ (\vdash \mathcal{D}) \qquad \text{(definitional closure)}$$

if $A \leftarrow B_1, \ldots, B_m$ is a definitional rule (= rule in $\mathcal{D}$) and σ is a substitution. This inference principle is straightforward, as a rule is always applied under a certain substitution.

$$\frac{\{\Gamma\sigma, \Delta \vdash C : B \leftarrow \Gamma \in \mathcal{D}, A = B\sigma\}}{A, \Delta \vdash C}\ (\mathcal{D} \vdash\ )$$

If $\mathcal{D}(A)$ ('the *defining conditions* of A') denotes $\{\Gamma : A \leftarrow \Gamma$ is a substitution instance of a definitional rule$\}$, this can be written as

$$\frac{\{\Gamma, \Delta \vdash C : \Gamma \in \mathcal{D}(A)\}}{A, \Delta \vdash C}\ (\mathcal{D} \vdash\ ) \qquad \text{(definitional reflection)}$$

i.e., definitional reflection is formulated exactly as in the propositional case. The application of $(\mathcal{D} \vdash\ )$ is subjected to the following proviso:

$$\mathcal{D}(A\sigma) \subseteq (\mathcal{D}(A))\sigma$$

[10] This notation is only needed for the renaming variables with fresh variables (in the context of 'standardizing apart'), so there is no need to deal with sequences rather than sets of variables.

As the converse inclusion is trivial, it may as well be formulated as $\mathcal{D}(A\sigma) = (\mathcal{D}(A))\sigma$. Note that this is a proviso for the applicability of definitional reflection, not a condition for a definition $\mathcal{D}$ to make sense.

The proviso ensures that for a substitution instance of A no defining rules have to be considered beyond those relevant for A itself. Consider, e.g., the definition

$$\left\{\begin{array}{lcl} P(x) & \leftarrow & Q \\ P(a) & \leftarrow & \end{array}\right.$$

If the proviso is disregarded, the sequent

$$P(x) \vdash Q$$

can be derived by applying $(\mathcal{D}\vdash)$ to $Q \vdash Q$, though $P(a)$ is derivable without Q being derivable (i.e., $P(x) \rightarrow Q$ is not admissible). The proviso also prevents the application of $(\mathcal{D}\vdash)$ in the presence of *extra variables* in the premisses of rules, i.e., variables occurring in the body but not in the head of a rule as in the following definition:

$$\left\{\begin{array}{lcl} P(b) & \leftarrow & Q(x) \\ Q(a) & \leftarrow & \end{array}\right.$$

Disregarding the proviso gives

$$P(b) \vdash Q(b)$$

by applying $(\mathcal{D}\vdash)$ to $Q(b) \vdash Q(b)$, though $P(b)$ is derivable without $Q(b)$ being derivable (i.e., $P(b) \rightarrow Q(b)$ is not admissible). In order to deal with such extra variables, one could in the basic logic introduce a binding device such as existential quantification and write the above rule for $P(b)$ as

$$P(b) \leftarrow \exists x Q(x)$$

Putting it in positive terms, the proviso guarantees that $(\mathcal{D}\vdash)$ is closed under substitution in the sense that any substitution instance of $(\mathcal{D}\vdash)$ is an application of $(\mathcal{D}\vdash)$ as well. This makes $(\mathcal{D}\vdash)$ particularly well-suited for computational purposes, as it enables *lifting*, i.e. the permutation of evaluation with substitution, which is essential for the computation of substitutions (bindings). In this way, definitional reflection is used in extensions of logic programming (see [1],[6]). At the same time, this means that for the justification of $(\mathcal{D}\vdash)$ one may argue as in the propositional case. Call an application of a rule $B_1, \ldots, B_n \rightarrow B$ *direct*, if it is applied under the empty substitution, i.e., if it leads from $B_1, \ldots, B_n$ to B. Suppose the rule $A, \Delta \rightarrow C$ is applied using its instance $A\sigma, \Delta\sigma \rightarrow C\sigma$, i.e., leading from $A\sigma$ and $\Delta\sigma$ to $C\sigma$. This can at the same time be understood as a direct application of the rule $A\sigma, \Delta\sigma \rightarrow C\sigma$. By the propositional variant of $(\mathcal{D}\vdash)$, the latter rule is admissible, if the rules in $\{\Gamma, \Delta\sigma \rightarrow C\sigma : \Gamma \in \mathcal{D}(A\sigma)\}$ are admissible. By closure under substitution, this is the same as $\{\Gamma, \Delta \rightarrow C : \Gamma \in (\mathcal{D}(A)\}\sigma$, whose rules are admissible by assumption, since the rules in $\{\Gamma, \Delta \rightarrow C : \Gamma \in \mathcal{D}(A)\}$ are admissible.

However, for the purely declarative reading, which underlies the admissibility interpretation, a more general principle of definitional reflection is available. The principle $(\mathcal{D}\vdash)_\omega$ of *generalized definitional reflection* runs as follows:

$$\frac{\{\Gamma\sigma, \Delta\sigma \vdash C\sigma : B \leftarrow \Gamma \in \mathcal{D}', \sigma = mgu(A, B)\}}{A, \Delta \vdash C}\ (\mathcal{D}\vdash)_\omega$$

where $\mathcal{D}'$ is a variant of $\mathcal{D}$, which has no variables in common with $A, \Delta \vdash C$, and where $\sigma = mgu(A, B)$ means that only one most general unifier of A and B is considered, if there exists one at all. This restriction guarantees that the number of premisses of $(\mathcal{D}\vdash)_\omega$ is at most the number of rules in $\mathcal{D}$.
This principle allows one to obtain

$$P(x) \vdash Q(x)$$

given that

$$\begin{cases} P(a) & \leftarrow & Q(a) \\ P(b) & \leftarrow & Q(b) \end{cases}$$

are the only clauses defining the predicate P. Obviously, $(\mathcal{D}\vdash)$ does not suffice to reach this result.

Generalized definitional reflection is easily justified by the admissibility interpretation. Suppose the rule $A, \Delta \rightarrow C$ is applied under a substitution θ, yielding $C\theta$ from $A\theta$ and $\Delta\theta$. Then this application can be eliminated as follows, depending on how the premiss $A\theta$ is obtained. $A\theta$ must be an instance $B\rho$ of a head B of a definitional rule $B \leftarrow \Gamma$. Obviously, $B \leftarrow \Gamma$ can be viewed as a member of a variant $\mathcal{D}'$ of $\mathcal{D}$ with variables standardized apart from those occurring elsewhere in the derivation. Then $\theta \cup \rho$ is a unifier of A and B, and $A\theta = A\sigma\tau, B\rho = B\sigma\tau$ for an *mgu* σ of A and B, and τ chosen such that $\sigma\tau = \theta \cup \rho$. Since by assumption, $\Gamma\sigma, \Delta\sigma \rightarrow C\sigma$ is already available as an admissible rule, using its instance $\Gamma\sigma\tau, \Delta\sigma\tau \rightarrow C\sigma\tau$ yields $C\theta$ as required.

It is easy to see that $(\mathcal{D}\vdash)_\omega$ comprises $(\mathcal{D}\vdash)$ as a special case. One just has to realize that σ is an *mgu* of A and B if $A = B\sigma$ and variables are standardized apart. Furthermore, due to the proviso $\mathcal{D}(A\sigma) \subseteq (\mathcal{D}(A))\sigma$, there is no further *mgu* of A with the head B of a definitional clause, so that $(\mathcal{D}\vdash)_\omega$ can be applied to the premiss set of $(\mathcal{D}\vdash)$.

Generalized definitional reflection is called $(\mathcal{D}\vdash)_\omega$ since it implicitly contains a step from instances to free variables, which is reminiscent of the ω-rule in arithmetic. In the above example, one implicitly passes from

$$\begin{array}{l} P(a) \ \vdash \ Q(a) \\ P(b) \ \vdash \ Q(b) \end{array}$$

which is obtained by propositional definitional reflection, to

$$P(x) \ \vdash \ Q(x)$$

In section 8 the idea of an underlying universalization principle (called ω-principle) is investigated in more detail.

To distinguish $(\mathcal{D}\vdash)$ verbally from $(\mathcal{D}\vdash)_\omega$, I shall sometimes speak of 'simple' in contradistinction to 'generalized' definitional reflection.

4. Lorenzen's inversion principle

Lorenzen was the first to explicitly formulate a general inversion principle. Unfortunately, his own formulations are not quite correct. A counterexample to the version given in the first edition of [11] (1955) was given by Hermes [7] (1959), who proposed a corrected formulation. Lorenzen's version in the second edition of [11] (1969) took only part of Hermes' critique into account and is still incorrect. Another formulation given by Lorenz [9] (1980) is also not fully correct. In the following, when speaking of Lorenzen's inversion principle, I mean the inversion principle in the reading given to it by Hermes. However, in order to compare it with definitional reflection, I assume the standard first-order term/formula structure to be given for atoms, not just a word structure as in Lorenzen.[11]

It is crucial that in Lorenzen, definitional rules and formulas in derivations may have variables in common, i.e. they need not be, and, for many purposes, must not be standardized apart (see section 5 for this point). As there might be such common variables, Lorenzen has to impose certain conditions on them. In the following, using Hermes' notation, I write $\overline{E}$ instead of $var(E)$ for the set of variables in E. For a substitution τ and a set of variables V, let $\tau \restriction V$ be the restriction of τ to V.

As mentioned above, Lorenzen's inversion principle is intended and formulated as an admissibility principle. Using as before the turnstile '$\vdash$' to express admissibility, it

[11] I cannot discuss in this paper the proper treatment of inversion for this general case (though some of the points raised by Hermes are related to it). This would require the theory of equational unification under associativity, which does not lead to uniquely determined most general unifiers, but only to (not necessarily finite) minimal complete sets of unifiers. See [2].

runs as follows. Suppose

$$\begin{cases} B_1 & \leftarrow & \Gamma_1 \\ & \vdots & \\ B_n & \leftarrow & \Gamma_n \end{cases}$$

are exactly those rules in a definition $\mathcal{D}$, which have joint instances with A, i.e., for which there are substitutions σ_i, ρ_i such that $A\sigma_i = B_i\rho_i$ for all i $(1 \leq i \leq n)$. Then the inversion principle can be expressed by the following inference schema:

$$\frac{\Gamma_1 \vdash C \quad \ldots \quad \Gamma_n \vdash C}{A \vdash C}\,(IP(\mathcal{D}))$$

provided the following two conditions are satisfied for all i $(1 \leq i \leq n)$:

(HC1) $\overline{\Gamma_i} \cap \overline{C} \subseteq \overline{A} \cap \overline{B_i}$
(HC2) If $A\sigma = B_i\rho$, then σ and ρ agree on $\overline{A} \cap \overline{B_i}$

As these conditions were proposed by Hermes [7], I call them the *Hermes conditions.*

That $IP(\mathcal{D})$ is correct with respect to the intended admissibility interpretation can be seen as follows: Consider an application of the rule

(R) $A \rightarrow C$

using the substitution σ, thus leading from $A\sigma$ to $C\sigma$. Then $A\sigma$ is derived by an application of the rule

(R_i) $\Gamma_i \rightarrow B_i$

in the last step, using some substitution ρ, thus leading from $\Gamma_i\rho$ to $B_i\rho$ where $B_i\rho = A\sigma$. This situation can be depicted as follows:

$$R\,\frac{A\sigma}{C\sigma} \;=\; \frac{\Gamma_i\rho}{B_i\rho}\,R_i$$

It can be assumed that $dom(\sigma) \subseteq \overline{A \rightarrow C}$ and $dom(\rho) \subseteq \overline{\Gamma_i \rightarrow B_i}$, i.e. σ and ρ are not defined for variables outside the respective rules. Let

$$\sigma' := \sigma \restriction C \quad \rho' := \rho \restriction \Gamma_i$$

Then by *HC1*

$$dom(\sigma') \cap dom(\rho') \;\subseteq\; \overline{C} \cap \overline{\Gamma_i} \;\subseteq\; \overline{A} \cap \overline{B_i}$$

By *HC2*, σ and ρ and therefore σ' and ρ' agree on $\overline{A} \cap \overline{B_i}$. Thus σ' and ρ' agree on $dom(\sigma') \cap dom(\rho')$, which means that

$$\tau := \sigma' \cup \rho'$$

is well defined and

$$\begin{aligned}&\Gamma_i\rho = \Gamma_i\rho' = \Gamma_i\tau\\&C\sigma = C\sigma' = C\tau\end{aligned}$$

Thus $\Gamma_i\tau \vdash C\tau$ is the same as $\Gamma_i\rho \vdash C\sigma$, which means that the given application of $A \rightarrow C$ (under the substitution σ) can be eliminated by using an application of $\Gamma_i \rightarrow C$ under the substitution τ, passing directly from $\Gamma_i\rho$ to $C\sigma$ and circumventing $A\sigma(= B_i\rho)$.

If in analogy with definitional reflection the inversion principle were formulated as including a context Δ:

$$\frac{\Gamma_1, \Delta \vdash C \quad \ldots \quad \Gamma_n, \Delta \vdash C}{A, \Delta \vdash C}$$

then *(HC1)* would have to be appropriately modified:

$(HC1')$ $(\overline{\Gamma_i} \cup \overline{\Delta}) \cap \overline{C} \subseteq \overline{A} \cap \overline{B_i}$ for all i $(1 \leq i \leq n)$

Digression: Other attempts at formulating the inversion principle

In the first edition of [11] (1955) Lorenzen formulated the inversion principle with the following variable condition:

(LC1) $\overline{B_i} \cap \overline{C} = \emptyset$ for all i $(1 \leq i \leq n)$

In the second edition of [11] (1969), though aware of Hermes' formulation, he chose instead

(LC2) $\overline{B_i} \subseteq \overline{A}$ for all i $(1 \leq i \leq n)$

Lorenz [9] (1980) proposed

(LC3) $\overline{\Gamma_i \rightarrow B_i} \subseteq \overline{A}$ for all i $(1 \leq i \leq n)$

The following counterexample shows that none of these conditions is appropriate. Let terms be a constant c as well as numerals of the form $0, s(0), s(s(0)), \ldots$ for another constant 0 and a unary function symbol s. Let P be a unary predicate. Let the definitional rules for P be as follows:

$$\begin{cases} \quad P(c) \ \leftarrow & \qquad\qquad (R_1) \\ P(s(0)) \ \leftarrow & P(x) \quad (R_2) \end{cases}$$

Obviously, the only derivable formulas are $P(c)$ and $P(s(0))$. Thus the rule

(R) $P(s(x)) \rightarrow P(x)$

is not admissible, as it would enable one to derive $P(0)$. Since R_2 is the only rule by means of which the premiss of an application of R can be derived,

$$\frac{P(x) \vdash P(x)}{P(s(x)) \vdash P(x)}$$

is an instance of the inversion principle, yielding the admissibility of R. As can easily be checked, each of the conditions *LC1, LC2* and *LC3* is satisfied, demonstrating that neither of them is appropriate to ensure the validity of the inversion principle, whereas the Hermes conditions block its application. Though *HC2* is satisfied, *HC1* does not hold, as

$$\overline{P(x)} \cap \overline{P(x)} = \{x\} \neq \emptyset = \overline{P(s(x))} \cap \overline{P(s(0))}$$

5. Standardizing apart

In logic programming, one normally assumes that variables in derivations are different from those occurring in the primitive rules of a program. As variables in rules are (semantically) understood as universally quantified, this can always be achieved by renaming, a procedure called *standardizing variables apart* (see [8]). This is also assumed in the formulation of generalized definitional reflection. More precisely, the inference schema $(\mathcal{D}\vdash)_\omega$ refers to an appropriate variant $\mathcal{D}'$ of $\mathcal{D}$, whose variables do not occur in the conclusion of $(\mathcal{D}\vdash)_\omega$. If this assumption is made in the case of $IP(\mathcal{D})$ by expecting the rules $B_i \leftarrow \Gamma_i$ to be variants of rules in $\mathcal{D}$ whose variables are different from those in $A \vdash C$, then the Hermes conditions *HC1* and *HC2* are trivially satisfied: $\overline{\Gamma_i} \cap \overline{C} = \emptyset$ yields *HC1*, and $\overline{A} \cap \overline{B_i} = \emptyset$ yields *HC2*.

This indicates that the inversion principles becomes too weak in this case. Consider the language with variables over an arbitrary domain, and the unary T as its only predicate (‘truth’). Extend the language with a binary function symbol $\wedge$ not yet available in the language and give it the definition

$$\mathcal{D} \left\{ \; T(x \wedge y) \;\; \leftarrow \;\; T(x), T(y) \quad (R) \right.$$

Then

$$\begin{array}{ll} T(x\wedge y) \rightarrow T(x) & (R_1) \\ T(x\wedge y) \rightarrow T(y) & (R_2) \end{array}$$

are admissible in the extended system. To demonstrate this using the inversion principle, one would use

$$\frac{T(x), T(y) \vdash T(x)}{T(x\wedge y) \vdash T(x)}$$

as an instance of $IP(\mathcal{D})$, for which the Hermes conditions are fulfilled. This instance establishes the admissibility of R_1, as its premiss $T(x), T(y) \vdash T(x)$ is trivially obtained[12] (analogously for R_2).
Now suppose that R is formulated with variables standardized apart:

$$\mathcal{D}' \left\{ \; T(u{\wedge}v) \;\; \leftarrow \;\; T(u), T(v) \quad (R') \right.$$

In order to establish the admissibility of R_1 by the inversion principle, the instance of $IP(\mathcal{D}')$ needed is

$$\frac{T(u), T(v) \vdash T(x)}{T(x{\wedge}y) \vdash T(x)}$$

which is valid (under the admissibility interpretation of sequents), but whose premiss $T(u), T(v) \vdash T(x)$ is simply not true.

This is a strong argument against Lorenzen's inversion principle as it stands. It is not invariant with respect to the formation of variants of definitional rules, i.e., the inferences generated by $IP(\mathcal{D})$ may differ from those generated by $IP(\mathcal{D}')$ for a variant $\mathcal{D}'$ of $\mathcal{D}$. Contrary to that, $(\mathcal{D}\vdash)_\omega$ is, by its very formulation, invariant in that respect, i.e., every inference step of the form $(\mathcal{D}\vdash)_\omega$ can be viewed as an inference step $(\mathcal{D}'\vdash\;)_\omega$ for any variant $\mathcal{D}'$ of $\mathcal{D}$.

This example gives a further clue why definitional reflection is preferable over Lorenzen's inversion principle. Let for $\mathcal{D}'$ the situation

$$R_1\,\frac{T(a{\wedge}b)\;=\;\dfrac{T(a) \quad T(b)}{T(a{\wedge}b)}\,R'}{T(a)}$$

be given, which can be analyzed as

$$R_1\,\frac{T(x{\wedge}y)\sigma\;=\;\dfrac{T(u)\rho \quad T(v)\rho}{T(u{\wedge}v)\rho}\,R'}{T(x)\sigma}$$

with $\rho = [{}^{u}/_{a}, {}^{v}/_{b}]$ and $\sigma = [{}^{x}/_{a}, {}^{y}/_{b}]$. Then, in order to circumvent the applications of R_1 and R' and to proceed directly from $T(u)\rho$ to $T(x)\sigma$, only the *substitution instances* $R'\rho$ and $R_1\sigma$ of the rules R' and R_1, respectively, are needed, which, if they are standardized apart, can be written as a single unifier $\theta = \rho \cup \sigma$. Moreover, in order to receive a principle which covers all possible substitutions, it is sufficient to consider *most general* unifiers.

The misconstruction of Lorenzen's inversion principle lies in the fact that, in order to establish the admissibility of $A \to C$, Lorenzen requires the admissibility of the *full* rules

[12] Here thinning is presupposed. In a substructural logic without thinning, the principle of definitional reflection and the inversion principle would have to formulated in such a way that in the antecedents of their premisses they only refer to single elements of Γ.

$\Gamma_i \rightarrow C$ as premisses of inversion, rather than the admissibility of certain substitution instances thereof. In expecting $\Gamma_i \vdash C$ rather than $\Gamma_i\theta \vdash C\theta$ (for an mgu θ of A and B_i) to hold, sophisticated variable conditions have to be imposed. For that Lorenzen cannot be blamed, as in the early 1950s, when he conceived his admissibility theory, the theory of unification as developed in the context of automated theorem proving (especially resolution), was far from available, Robinson's 1965 paper [17] being the key publication. Relying on the modern theory of substitution and unification, I would claim that generalized definitional reflection with its way of considering most general unifiers retains most, if not all, intentions of Lorenzen's inversion principle[13].

6. The inversion principle follows from generalized definitional reflection

I now show that $IP(\mathcal{D})$ can be formally derived from $(\mathcal{D}\vdash)_\omega$, using only the principle of substitution

$$\frac{\Gamma \vdash A}{\Gamma\sigma \vdash A\sigma}\ \text{(Subst)}$$

together with elementary properties of substitutions. More precisely, I show the following: Read sequents as formal expressions. Let $\mathcal{D}$ be any definition. Consider a sequent system based on the inference schemata of (Subst) and $(\mathcal{D}\vdash)_\omega$. Suppose a rule $A \rightarrow C$ is given. Write $\mathcal{D}$ as

$$\mathcal{D}\left\{\begin{array}{ccc} & \vdots & \\ B_1 & \leftarrow & \Gamma_1 \\ & \vdots & \\ B_n & \leftarrow & \Gamma_n \\ & \vdots & \end{array}\right.$$

such that the displayed rules $B_i \leftarrow \Gamma_i$ $(1 \leq i \leq n)$ are exactly those definitional rules in $\mathcal{D}$ whose heads have joint instances with A, i.e., there are σ_i, ρ_i such that $A\sigma_i = B_i\rho_i$. Suppose *HC1* and *HC2* are satisfied. Then $IP(\mathcal{D})$ is a derivable inference schema of this sequent system, i.e., $A \vdash C$ can be derived from $\{\Gamma_i \vdash C_i : 1 \leq i \leq n\}$.

As Lorenzen's formulation of the inversion principle does not use contexts Δ, I restrict myself to the versions of $(\mathcal{D}\vdash)$ and $IP(\mathcal{D})$ without such contexts. For the proof with contexts, which is a straightforward generalization of the one given here,

[13] Of course, again, apart from its restriction to full term structure instead of Lorenzen's general word structure.

one has to use the extended Hermes condition *HC1′*. So the schemata to be compared are:

$$\frac{\{\Gamma'_i\sigma_i \vdash C\sigma_i : \sigma_i = mgu(A, B'_i), 1 \leq i \leq n\}}{A \vdash C}\ (\mathcal{D}\vdash)_\omega$$

where for each i $(1 \leq i \leq n)$, $B'_i \leftarrow \Gamma'_i$ is a variant of $B_i \leftarrow \Gamma_i$ which has no variables in common with $A \vdash C$, and

$$\frac{\{\Gamma_i \vdash C : 1 \leq i \leq n\}}{A \vdash C}\ IP(\mathcal{D})$$

The following proof constructs, for each i $(1 \leq i \leq n)$, a variant $B'_i \leftarrow \Gamma'_i$ of $B_i \leftarrow \Gamma_i$, which has no variables in common with $A \vdash C$, such that for $\sigma_i := mgu(A, B'_i)$, the sequent $\Gamma'_i\sigma_i \vdash C\sigma_i$ is a substitution instance of $\Gamma_i \vdash C$. This means that the premisses of $(\mathcal{D}\vdash)_\omega$ are obtained from those of $IP(\mathcal{D})$ by substitution.
Define the following sets of variables:

$$\overline{x}_i := \overline{A} \cap \overline{B_i}$$
$$\overline{x}'_i := \overline{\Gamma_i \to B_i} \cap \overline{A \to C}$$
$$\overline{x}^*_i := \overline{x}'_i \setminus \overline{x}_i$$

Corresponding to these sets, generate disjoint sets of fresh variables $\overline{y}_i, \overline{y}'_i, \overline{y}^*_i$ of the same cardinalities, such that

$$B_i[\overline{x}'_i/\overline{y}'_i] \leftarrow \Gamma_i[\overline{x}'_i/\overline{y}'_i]$$

is a variant of $B_i \leftarrow \Gamma_i$ which has no variables in common with $A \vdash C$. This variant is abbreviated as $B'_i \leftarrow \Gamma'_i$. Let $\sigma_i := mgu(A, B'_i)$. Then

$$A\sigma_i = B_i[\overline{x}'_i/\overline{y}'_i]\sigma_i = B_i[\overline{x}^*_i/\overline{y}^*_i][\overline{x}_i/\overline{y}_i]\sigma_i$$

By *HC2*, the substitutions σ_i and $[\overline{x}^*_i/\overline{y}^*_i][\overline{x}_i/\overline{y}_i]\sigma_i$ agree on $\overline{x}_i$, which means that

$$(*)\quad [\overline{x}_i/\overline{y}_i]\sigma_i = \sigma_i$$

since $\overline{x}_i$ and $\overline{x}^*_i$ are disjoint, and outside $\overline{x}_i$ the substitutions σ_i and $[\overline{x}_i/\overline{y}_i]\sigma_i$ trivially agree.
Now $\Gamma_i \vdash C$ has

$$\Gamma'_i \vdash C[\overline{x}'_i/\overline{y}'_i]$$

as a substitution instance, which can also be written as

$$\Gamma'_i \vdash C[\overline{x}^*_i/\overline{y}^*_i][\overline{x}_i/\overline{y}_i]$$

By *HC1*, those variables of $\overline{x}_i^*$, which occur in C, do not occur in $\overline{\Gamma'_i}$. Therefore renaming them back, i.e., substituting with $[\overline{y}_i^*/\overline{x}_i^*]$, yields

$$\Gamma'_i \vdash C[\overline{x}_i/\overline{y}_i]$$

Substitution with σ_i gives

$$\Gamma'_i\sigma_i \vdash C[\overline{x}_i/\overline{y}_i]\sigma_i$$

from which, by $(*)$,

$$\Gamma'_i\sigma_i \vdash C\sigma_i$$

is obtained as desired.

That *IP*$(\mathcal{D})$ is *strictly* weaker than $(\mathcal{D}\vdash)_\omega$ follows by using the results of section 5 on standardizing rules apart, since, given the definitional rule

$$T(u\wedge v) \leftarrow T(u), T(v)$$

the sequent $T(x\wedge y) \vdash T(x)$ is derivable by $(\mathcal{D}\vdash)_\omega$, but not by *IP*$(\mathcal{D})$. Another example, which does not rely on the problems the inversion principle has with the renaming of variables, is the following. Let P and Q be unary predicates and a the only constant available. Suppose

$$\mathcal{D}\ \left\{\begin{array}{ccccc} & & \vdots & & \\ P(a) & \leftarrow & Q(a) & & (R_1) \\ & & \vdots & & \end{array}\right.$$

has R_1 as the only clause defining P. Then

$$\frac{Q(a) \vdash Q(a)}{P(x) \vdash Q(x)}$$

is an instance of $(\mathcal{D}\vdash)_\omega$ with the premiss trivially derivable, whereas a corresponding instance of *IP*$(\mathcal{D})$ leading to $P(x) \vdash Q(x)$ is

$$\frac{Q(a) \vdash Q(x)}{P(x) \vdash Q(x)}$$

whose premiss is not derivable.

7. Simple definitional reflection and the inversion principle are mutually independent

Both simple definitional reflection $(\mathcal{D}\vdash)$ and the inversion principle $IP(\mathcal{D})$ are strictly weaker than generalized definitional reflection $(\mathcal{D}\vdash)_\omega$. The following examples show that neither of them entails the other one without assuming further inference principles. In this sense they are independent of each other.

To show that $(\mathcal{D}\vdash)$ does not imply $IP(\mathcal{D})$ consider the definition

$$\begin{cases} P(a) & \leftarrow & P(a) \\ P(b) & \leftarrow & P(b) \\ Q & \leftarrow & P(a) \\ Q & \leftarrow & P(b) \end{cases}$$

Then using the inversion principle, $P(x)\vdash Q$ is obtained via

$$\frac{P(a)\vdash Q \quad P(b)\vdash Q}{P(x)\vdash Q}\ IP(\mathcal{D})$$

However, $P(x)\vdash Q$ cannot be derived using $(\mathcal{D}\vdash)$ as $P(x)$ is not an instance of the head of a definitional rule (apart from the fact that the proviso $\mathcal{D}(P(x)\sigma) \subseteq (\mathcal{D}(P(x)))\sigma$ is not satisfied).

To show that $IP(\mathcal{D})$ does not imply $(\mathcal{D}\vdash)$, one could argue as in the previous section that, given the definitional rule

$$T(u\wedge v) \leftarrow T(u), T(v)$$

the sequent $T(x\wedge y)\vdash T(x)$ is derivable by $(\mathcal{D}\vdash)$ (not only by $(\mathcal{D}\vdash)_\omega$), but not by $IP(\mathcal{D})$. An example not related to the 'standardizing apart'-problems of $IP(\mathcal{D})$ is the following. Consider the definition

$$\begin{cases} P(x) & \leftarrow & Q(x) \\ R & \leftarrow & Q(a) \\ Q(a) & \leftarrow & Q(a) \\ Q(b) & \leftarrow & Q(b) \end{cases}$$

Then using (simple) definitional reflection, $P(a)\vdash R$ can be derived via

$$\frac{Q(a)\vdash R}{P(a)\vdash R}\ (\mathcal{D}\vdash)$$

However, $P(a)\vdash R$ cannot be derived using $IP(\mathcal{D})$, since for that $Q(x)\vdash R$ would be needed as a premiss. In order to derive $Q(x)\vdash R$ by using $IP(\mathcal{D})$, both $Q(a)\vdash R$ and $Q(b)\vdash R$ must be derivable. But $Q(b)\vdash R$ is not derivable.

8. Universalization: The ω-principle

Generalized definitional reflection $(\mathcal{D}\vdash)_\omega$ is equivalent to simple definitional reflection $(\mathcal{D}\vdash)$, if the following principle is added, called the *ω-principle*:

$$\frac{\{A\sigma, \Delta\sigma \vdash C\sigma : B \leftarrow \Gamma \in \mathcal{D}', \sigma = mgu(A,B)\}}{A, \Delta \vdash C}\ (\omega)_{\mathcal{D}}$$

As with $(\mathcal{D}\vdash)_\omega$, $\mathcal{D}'$ is a variant of $\mathcal{D}$, which has no variables in common with $A, \Delta \vdash C$, and $\sigma = mgu(A,B)$ means that only one most general unifier of A and B is considered. The difference between $(\mathcal{D}\vdash)_\omega$ and $(\omega)_{\mathcal{D}}$ is that the latter allows one to pass from all defined substitution instances of a sequent to the sequent itself, without replacing A on the left side with its defining conditions. In this way the basic idea of definitional reflection is separated from the *universalization step* contained in passing from $\{A\sigma, \Delta\sigma \vdash C\sigma : \ldots\}$ to $A, \Delta \vdash C$.

It is easy to see that, as an admissibility principle, $(\omega)_{\mathcal{D}}$ is valid. Suppose the rule $A, \Delta \rightarrow C$ is applied using its substitution instance $A\sigma, \Delta\sigma \rightarrow C\sigma$, leading from $A\sigma$ and $\Delta\sigma$ to $C\sigma$. Then the premiss $A\sigma$ is derived using an instance $B\rho \leftarrow \Gamma\rho$ of a definitional rule $B \leftarrow \Gamma$. Since it is assumed that variables are standardized apart, $\rho \cup \sigma$ is well defined and is a unifier of A and B. Furthermore, $(\Delta\sigma, A\sigma \rightarrow C\sigma) = (\Delta(\sigma \cup \rho), A(\sigma \cup \rho) \rightarrow C(\sigma \cup \rho))$. Since $\rho \cup \sigma = \theta\tau$ for $\theta = mgu(A,B)$, the admissibility of $\Delta\sigma, A\sigma \rightarrow C\sigma$ follows by substitution from that of $\Delta\theta, A\theta \rightarrow C\theta$, which is assumed as a premiss of $(\omega)_{\mathcal{D}}$.

To see that $(\omega)_{\mathcal{D}}$ is a consequence of $(\mathcal{D}\vdash)_\omega$, one may use thinning, passing in its premiss set from

$$A\sigma, \Delta\sigma \vdash C\sigma$$

to

$$\Gamma\sigma, A\sigma, \Delta\sigma \vdash C\sigma$$

then use $(\mathcal{D}\vdash)_\omega$ yielding

$$A, A, \Delta \vdash C$$

and then use contraction. Another possible strategy would be to apply cut using the sequent

$$\Gamma\sigma \vdash A\sigma$$

which can be obtained using $(\vdash\mathcal{D})$.

Conversely, in order to show that $(\mathcal{D}\vdash)_\omega$ is derivable from $(\vdash\mathcal{D})$ together with $(\omega)_{\mathcal{D}}$, consider an application of $(\mathcal{D}\vdash)_\omega$ with conclusion $A, \Delta \vdash C$. Suppose $A\sigma = B\sigma$ for a rule $B \leftarrow \Gamma$ in $\mathcal{D}$ with $\sigma = mgu(A,B)$. Suppose $A\sigma = B'\rho$ for a rule $B' \leftarrow \Gamma'$ in $\mathcal{D}$.

Since variables are assumed to be standardized apart, $\sigma \cup \rho$ is a unifier of A and B'. Let $\theta := mgu(A, B')$. Then $\Gamma'\theta, \Delta\theta \vdash C\theta$ is a premiss of $(\mathcal{D}\vdash)_\omega$, from which by substitution $\Gamma'(\sigma \cup \rho), \Delta(\sigma \cup \rho) \vdash C(\sigma \cup \rho)$ is obtained, which is the same as $\Gamma'\rho, \Delta\sigma \vdash C\sigma$. Therefore

$$M_\sigma := \{\Gamma\rho, \Delta\sigma \vdash C\sigma : A\sigma = B\rho \text{ for some } B \leftarrow \Gamma \text{ in } \mathcal{D}\}$$

is a subset of the set of premisses of $(\mathcal{D}\vdash)_\omega$. From M_σ one derives $A\sigma, \Delta\sigma \vdash C\sigma$ using $(\mathcal{D}\vdash)$. As this sequent is obtained for every $A\sigma$ in the premisses of $(\omega)_\mathcal{D}$, applying $(\omega)_\mathcal{D}$ yields $A, \Delta \vdash C$ as desired.

This result shows that $(\mathcal{D}\vdash)_\omega$ can be split up into $(\vdash\mathcal{D})$ and (ω). Generalized definitional reflection combines two ideas, which may be separated from each other. $(\mathcal{D}\vdash)$ expresses the step of definitional reflection, which refers to the definitional conditions of a formula, whereas $(\omega)_\mathcal{D}$ is a principle of *universalization*, leading from substitution instances of a sequent to the sequent itself. In this sense $(\omega)_\mathcal{D}$ can be viewed as being dual to substitution (Subst), which is a *specialization* principle. According to this picture there are two separate fundamental pairs of principles: Specialization (Subst) and universalization $(\omega)_\mathcal{D}$ for the handling of variables, and definitional closure $(\vdash\mathcal{D})$ and definitional reflection $(\mathcal{D}\vdash)$ for handling the relationship between definienda and defining conditions.

I consider this separation of universalization and definitional reflection to be a great conceptual advantage. This advantage is reflected by the fact that, when it comes to exploit the dualities between left and right inferences in the sequent calculus, one would treat $(\omega)_\mathcal{D}$ separate from the duality given by $(\vdash\mathcal{D})$ and $(\mathcal{D}\vdash)$. From the perspective of cut elimination, it is *simple* definitional reflection, not generalized definitional reflection, which complements the rule of definitional closure. There is no straightforward main reduction for a $(\vdash\mathcal{D})/(\mathcal{D}\vdash)_\omega$ cut, as side formulas in $(\mathcal{D}\vdash)_\omega$ do not remain unchanged between the premisses and the conclusion of $(\mathcal{D}\vdash)_\omega$, but there is an obvious main reduction for $(\vdash\mathcal{D})/(\mathcal{D}\vdash)$ cuts (see [19]). 'Harmony' or 'symmetry' obtains between $(\vdash\mathcal{D})$ and $(\mathcal{D}\vdash)$, not between $(\vdash\mathcal{D})$ and $(\mathcal{D}\vdash)_\omega$.

For these reasons, I prefer treating $(\omega)_\mathcal{D}$ as an elementary principle of reasoning rather than mixing it up with other ideas as in $(\mathcal{D}\vdash)_\omega$. Actually, as a principle of its own it is already quite powerful. For example, it suffices to derive the full theory of free equality. Given the definition

$$\left\{\; x \doteq x \;\leftarrow \right.$$

consisting of a single rule for the binary equality predicate $\doteq$, the following inference principles can be generated:

$$\frac{}{t \doteq t} \qquad \frac{\Gamma\sigma \vdash A\sigma}{s \doteq t, \Gamma \vdash A}\; \sigma = mgu(s,t) \qquad \frac{}{s \doteq t, \Gamma \vdash A}\; s \text{ and } t \text{ not unifiable}$$

which again are equivalent to the standard theory of free equality (see [21]).

In the presence of $(\omega)_{\mathcal{D}}$, $IP(\mathcal{D})$ and $(\mathcal{D}\vdash)$ are no longer mutually independent. Now the inversion principle is strictly weaker than simple definitional reflection. The result that, given $(\omega)_{\mathcal{D}}$, $(\mathcal{D}\vdash)$ entails $IP(\mathcal{D})$, follows from the fact that, as shown before, $(\mathcal{D}\vdash)_\omega$ is a consequence of $(\mathcal{D}\vdash)$ together with $(\omega)_{\mathcal{D}}$, and $IP(\mathcal{D})$ is a consequence of $(\mathcal{D}\vdash)_\omega$. However the converse does not hold. Consider again the example

$$\begin{cases} P(x) & \leftarrow & Q(x) \\ R & \leftarrow & Q(a) \\ Q(a) & \leftarrow & Q(a) \\ Q(b) & \leftarrow & Q(b) \end{cases}$$

already studied above. Even if $(\omega)_{\mathcal{D}}$ is available, the sequent $P(a)\vdash R$, which is derivable by means of $(\mathcal{D}\vdash)$, cannot be derived using $IP(\mathcal{D})$ and $(\omega)_{\mathcal{D}}$. As mentioned above, $Q(x)\vdash R$ is needed to derive $P(a)\vdash R$ using $IP(\mathcal{D})$, and in order to derive $Q(x)\vdash R$ by $(\omega)_{\mathcal{D}}$, both $Q(a)\vdash R$ and $Q(b)\vdash R$ must be available. But $Q(b)\vdash R$ is underivable. Actually, if the two premisses $Q(a)\vdash R$ and $Q(b)\vdash R$ were derivable, one would receive $Q(x)\vdash R$ already by $IP(\mathcal{D})$, as

$$\frac{Q(a)\vdash R \quad Q(b)\vdash R}{Q(x)\vdash R}$$

can be read both as an application of $(\omega)_{\mathcal{D}}$ and of $IP(\mathcal{D})$. This illustrates that $IP(\mathcal{D})$ itself incorporates certain ideas underlying $(\omega)_{\mathcal{D}}$, to the effect that in the present example, $(\omega)_{\mathcal{D}}$ does not add any deductive power to $IP(\mathcal{D})$.

Summarizing the results of this paper, both definitional reflection and the inversion principle are consequences of generalized definitional reflection. Both are strictly weaker than generalized definitional reflection, but neither of them entails the other one. If the ω-principle is added, the inversion principle is strictly weaker than simple definitional reflection (which is then equivalent to generalized definitional reflection). Whether there is some other general principle similar to $(\omega)_{\mathcal{D}}$, on the basis of which the inversion principle would entail definitional reflection and perhaps even generalized definitional reflection, remains to be investigated.

References

1. M. Aronsson, L.-H. Eriksson, A. Gäredal, L. Hallnäs & P. Olin, The programming language GCLA: A definitional appraoch to logic programming. *New Generation Computing*, **4** (1990), 381–404.

2. F. Baader & K. U. Schulz, Unification theory. In: W. Bibel, P.H. Schmitt (eds.), *Automated Deduction: A Basis for Applications*, Vol. I, Dordrecht: Kluwer 1998, 225–263.

3. G. Gentzen, Untersuchungen über das logische Schließen. *Mathematische Zeitschrift*, **39** (1935), 176–210, 405–431, English translation in: M.E. Szabo (ed.), *The Collected Papers of Gerhard Gentzen*, Amsterdam: North Holland 1969, 68–131.

4. L. Hallnäs, Partial Inductive Definitions. *Theoretical Computer Science* **87** (1991), 115–142.

5. L. Hallnäs, On the proof-theoretic foundation of general definition theory. In: R. Kahle & P. Schroeder-Heister (eds.), *Proof-Theoretic Semantics.* Special issue of *Synthese*, **148**(3)(2006), 589–602.

6. L. Hallnäs & P. Schroeder-Heister, 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.

7. H. Hermes, Zum Inversionsprinzip der operativen Logik. In: A. Heyting (ed.), *Constructivity in Mathematics*, Amsterdam: North-Holland, 1961, 62–68.

8. J. W. Lloyd, *Foundations of Logic Programming.* Berlin etc.: Springer, 2nd edition 1987.

9. K. Lorenz, Logik, operative. Entry in: J. Ritter & K. Gründer (eds.), *Historisches Wörterbuch der Philosophie*, Vol. 5, Basel: Schwabe, 1980, 444–452.

10. P. Lorenzen, Konstruktive Begründung der Mathematik. *Mathematische Zeitschrift*, **53** (1950), 162–201.

11. P. Lorenzen, *Einführung in die operative Logik und Mathematik.* Springer: Berlin 1955, 2nd ed. Berlin 1969.

12. E. Moriconi & L. Tesconi, On inversion principles. Manuscript (submitted for publication).

13. D. Prawitz, *Natural Deduction: A Proof-Theoretical Study*, Stockholm: Almqvist & Wiksell 1965. Reprinted Mineola, N.Y.: Dover Publ., 2006.

14. D. Prawitz, On the idea of a general proof theory, *Synthese* **27** (1974), 63–77.

15. D. Prawitz, Proofs and the meaning and completeness of the logical constants. In: J. Hintikka et al. (eds.), *Essays on Mathematical and Philosophical Logic*, Dordrecht: Reidel 1978, 25–40 (revised German translation in *Conceptus* **16** (1982)).

16. D. Prawitz, Meaning approached via proofs. In: R. Kahle & P. Schroeder-Heister (eds.), *Proof-Theoretic Semantics*. Special issue of *Synthese*, **148**(3)(2006), 507–524.

17. J. A. Robinson, A machine-oriented first-order logic. Abstract. *Journal of Symbolic Logic. 28,* 1963, p. 302. Full paper: A machine-oriented logic based on the resolution principle, *Journal of the Association for Computing Machinery* **12** (1965), 23–41.

18. P. Schroeder-Heister, A natural extension of natural deduction, *Journal of Symbolic Logic* **49** (1984), 1284–1300.

19. P. Schroeder-Heister, 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.

20. P. Schroeder-Heister, Cut Elimination in Logics with Definitional Reflection, in: D. Pearce & H. Wansing (eds.), *Nonclassical Logics and Information Processing. International Workshop, Berlin, November 1990, Proceedings.* Berlin: Springer LNCS, Vol. 619, 1992, 146–171.

21. P. Schroeder-Heister, Definitional reflection and the completion. In: R. Dyckhoff (ed.), *Extensions of Logic Programming. Proceedings of the 4th International Workshop, ELP '93, St. Andrews, March/April 1993.* Berlin: Springer LNCS, Vol. 798, 1994, 333–347.

22. P. Schroeder-Heister, On the notion of *assumption* in logical systems. In: R. Bluhm & C. Nimtz (eds.), *Selected Papers Contributed to the Sections of GAP5, Fifth International Congress of the Society for Analytical Philosophy, Bielefeld, 22-26 September 2003*, Paderborn: mentis, 2004, 27–48.

23. P. Schroeder-Heister, Validity concepts in proof-theoretic semantics. In: R. Kahle & P. Schroeder-Heister (eds.), *Proof-Theoretic Semantics.* Special issue of *Synthese*, **148**(3) (2006), 525–571.

24. P. Schroeder-Heister, Lorenzens operative Logik und moderne beweistheoretische Semantik. In: J. Mittelstraß (ed.), *Der Konstruktivismus in der Philosophie im Ausgang von Wilhelm Kamlah und Paul Lorenzen* (Beiträge zum Symposium in Konstanz, September 2005). 2007 (in print).

25. P. Schroeder-Heister, Operative logic and proof-theoretic semantics. In: E.H. Haeusler & L.C. Pereira (eds.), *Advances in Natural Deduction* (volume under review).