Notre Dame Journal of Formal Logic
Volume 37, Number 3, Summer 1996

A Constructive Valuation Semantics for Classical Logic

FRANCO BARBANERA and STEFANO BERARDI

Abstract This paper presents a constructive interpretation for the proofs in classical logic of Σ^0_1-sentences and for a witness extraction procedure based on Prawitz's reduction rules.

1 Introduction Cut-elimination theorems play a fundamental role in proof theory. Many relevant properties of logics can be derived from them. In particular, for intuitionistic logic, cut-elimination allows one to "compute" with proofs. In fact, the constructive contents of an intuitionistic proof can be made *explicit* by eliminating its cuts. In the natural deduction version of intuitionistic logic Prawitz [8] cut-elimination corresponds to normalizability of proofs, that is, to the possibility of getting rid of any detour by means of suitable reduction rules. (A detour is an application of an introduction rule for a connective immediately followed by its corresponding elimination rule.) Such reduction rules preserve the well-known functional interpretation of intuitionistic connectives and proofs which Brouwer, Heyting, Kolmogorov, and others proposed in order to allow a better understanding of the constructive features of intuitionistic logic (see for BHK interpretation, Kolmogorov [6] and Heyting [4]). Since irreducible proofs explicitly represent mathematical constructions, reduction rules for intuitionistic logic turn out to have a *computational* meaning.

The BHK interpretation was also helpful to the development of typed functional languages and of computer science in general, for instance, through the so-called Curry-Howard analogy. The understanding of intuitionistic logic provided by the BHK interpretation also enabled the development of simple and comprehensible procedures of program extraction from formal constructive proofs. Such procedures at the very beginning of their development some thirty years ago were extremely involved. By interpreting the implication as a function space constructor it was possible to interpret a proof of the proposition (type) $A \rightarrow B$ as a recursive map from proofs of proposition (type) A to those of B, and the reduction rule for implication in terms of the β-rule for λ-calculus.

Received March 4, 1994; revised November 28, 1995

One of the main results derived from the cut-elimination theorem for intuitionistic logic, and hence from normalizability in natural deduction, is the consistency of the logical system. In the natural deduction for classical logic, instead, one cannot get any consistency from normalizability, unless the set of reduction rules for intuitionistic logic is suitably extended. So Prawitz added to the set of the intuitionistic reduction rules a number of reductions that transform a classical proof into one in which eliminations of double negations are performed only on atomic propositions [8]. Working on the *weak* normalization property of this enlarged set of rules (proved in [8]), Prawitz managed to get the consistency for classical logic as well. As for what concerns the *strong* normalization property, a sketchy proof was given by Prawitz in [9], while a complete and detailed proof can be found in Barbanera and Berardi [1]. The use of the new reductions for classical logic was merely technical and no interpretation was provided for them.

There is another interesting use of the above mentioned extended set of reductions: a *computational* one. Even if it can hardly be thought of as a realistic one for classical logic, which is typically nonconstructive, the possibility of such a use has been known for a long time. In [5], Kreisel, by means of his no-counterexample interpretation for classical proofs, showed that classical and intuitionistic provability coincide if we consider only Σ^0_1-sentences. Later, Friedman [3] enforced Kreisel's result by providing a translation from classical to intuitionistic proofs of Σ^0_1-sentences. So the outcome of the above mentioned results is that classical logic also has computational features. However, in order to reach a full understanding of it and to be able to use such computational features in practice, one should also devise methods to *directly* extract the constructive contents of proofs of Σ^0_1-sentences, something not provided by Kreisel's and Friedman's results. It would be desirable for normalizability for natural deduction to provide such a method for classical logic. In fact, it is like this: in [1] it was proved that, by normalizing classical proofs with respect to Prawitz's reductions for classical logic, one can manage to exploit Kreisel's no-counterexample interpretation, and extract the constructive contents, that is, witnesses, from classical proofs of Σ^0_1-sentences. By allowing free variables in proofs it is also easy to get functions out of classical proofs of Π^0_2-sentences.

The *extraction by normalization* method of [1], as recalled in Section 3, however, cannot be counted as a real success in the improvement of the understanding of the constructive features of classical logic, unless it can be explained in the setting of a clear computational interpretation of classical natural deduction; in particular, of Prawitz's reductions for classical logic. Providing such an interpretation is what this paper attempts. As a matter of fact, any technical mathematical result cannot really provide any actual improvement in the understanding of a topic until it is correctly *interpreted*.

In the classical case, the BHK interpretation for formulas and proofs cannot be applied. Let us see why. By the fact that the BHK interpretation of the intuitionistic falsehood ($\perp$) is the empty set, we get that the interpretation of $\neg A$ ($\equiv A \to \perp$) is empty in the case in which the interpretation of A is inhabited. If, instead, the interpretation of A is empty, then the interpretation of $\neg A$ consists of the sole function from the empty set to the empty set. Then it follows that the interpretation of $\neg\neg A (\equiv \neg A \to \perp)$ consists of at most one element. Any function from a set with at most one element is constant. Thus the classical axiom $\neg\neg A \rightarrow A$ should be interpreted, from BHK's viewpoint, as a *constant* function from (the interpretation of) $\neg\neg A$ to (the interpretation of) A, that is, as *an effective way to pick an element of a nonempty A, uniformly on A.* One can easily see, however, that such a choice function can hardly be computable.

Then a constructive interpretation of classical logic necessarily must give a meaning different from BHK's one to the classical falsehood $\perp$. The connectives $\wedge$ and $\forall$ can instead be given the BHK interpretation, while the connectives $\neg$, $\rightarrow$, $\vee$ and $\exists$ can be defined from the previous ones, once an interpretation of $\perp$ is provided.

In order to find an interpretation for the classical $\perp$ we ask ourselves the following question: What is the role of $\perp$ in classical proofs? While intuitionistic logic is essentially concerned with the notion of provability, classical logic deals with the notion of truth. Hence, in a proof

$$\begin{array}{c} A_1\ A_2 \ldots A_n \\ \vdots \\ \perp \end{array} \tag{1}$$

the meaning of $\perp$ is that it is not possible to assume consistently all the premises $A_1, A_2, \ldots, A_n$, that is, that any model satisfying all of such formulas cannot be a consistent one. This remains valid even through double negation elimination. In other words (1) can be seen as a particular way to build inconsistent models out of models for the formulas $A_1, A_2, \ldots, A_n$. Hence $\perp$ could rightly be interpreted as the set of all inconsistent models. We can be even more general and consider models not completely specified (partial models henceforth). From this more general point of view, the interpretation of $\perp$ turns out to be the set of all inconsistent *partial* models.

Once given this interpretation for $\perp$ we can extend it to all the formulas. A formula A can be seen as the set of the partial models satisfying it and, as for $\perp$, a proof of it is a way of building one of these models. Conjunction is now an operator on partial models: if v_1 is an element of A_1 and v_2 of A_2, then $v_1 + v_2$ (the "union" of the models) is an element of $A_1 \wedge A_2$. The elements of a negation $\neg A$ are now the ones which cannot be elements of A, that is, those that turn out to belong to the interpretation of $\perp$ once they are put together with elements of A. In this way the negation turns out to have a Kripke semantics. In a sense, an element of $\neg A$ can be seen as a *function* that returns an element of $\perp$ when it is applied to an element of A. That is, $\neg A$ can be also considered as $A \rightarrow \perp$, giving to "$\rightarrow$" its BHK interpretation.

We shall formalize the above interpretation in a way similar to that in which Martin-Löf formalized the computational interpretation of constructive logic into his Type Theory [7]. So we will define a system, *VS*, for classical logic which is essentially an elaboration of usual classical natural deduction. The rules of *VS* describe, for each connective, how to *build* and *use* the elements belonging to the interpretation of formulas having the considered connective as the main one. As in Martin-Löf's Type Theory, our system will deal with judgments about logical formulas and elements of their interpretation, that is, partial models. A partial model can be seen as a set of 'assertions' stating the truth ($\mathbf{t}.P$) or falsity ($\mathbf{f}.P$) of closed atomic formulas. Such sets of assertions will be called *valuations* and denoted by *valuation terms*: $\mathsf{v}, \mathsf{v}', \ldots$. Formally, our judgments will be expressions of the form $\mathsf{v} \models \gamma.A$ (γ being $\mathbf{t}$

or **f**). A constructive interpretation, stronger than the natural "the valuation (denoted by v) *forces* $\gamma.A$," will be provided for such judgments. In particular, we shall interpret

$\mathsf{v} \models \mathbf{t}.P$, for P atomic	as	it is possible to get a *finite* subvaluation of v such that the conjunction of its assertions implies $\mathbf{t}.P$.

The remaining cases will be interpreted as in Kripke models, that is,

$\mathsf{v} \models \mathbf{t}.\forall x.A(x)$	as	for any t it is possible to get (the interpretation of) $\mathsf{v} \models \mathbf{t}.A(t)$,
$\mathsf{v} \models \mathbf{t}.A \wedge B$	as	(the interpretation of) [$\mathsf{v} \models \mathbf{t}.A$ *and* $\mathsf{v} \models \mathbf{t}.B$], and
$\mathsf{v} \models \mathbf{f}.A$	as	it is possible to get a procedure that, whenever given a valuation v' validating $\mathbf{t}.A$, returns a *finite* inconsistent subvaluation of $\mathsf{v} + \mathsf{v}'$.

Moreover, the logical rules can be interpreted in such a way that a proof in normal form of a judgment effectively provides the constructive interpretation for the judgment.

For Σ^0_1-sentences this interpretation provides a means to get witnesses. A closed derivation for a formula $\exists x.P(x)$ in our system becomes a derivation for the judgment $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$, $\varnothing$ denoting the empty valuation. Hence, a proof of such a judgment can be interpreted as a function that takes a valuation v_0 validating $\mathbf{t}.\forall x.\neg P(x)$ and returns an inconsistent subvaluation of v_0. If we managed to have v_0 be the valuation consisting of all possible assertions $\mathbf{f}.P(t)$, we could get a finite inconsistent subvaluation $\{\mathbf{f}.P(t_1), \ldots, \mathbf{f}.P(t_n)\}$. From that, we can obtain some *witnesses* for $\exists x.P(x)$, that is, some t_i's such that $P(t_i)$ holds.

We shall see that the witness extraction method, based on Prawitz's reductions and devised in [1], will provide a means to "feed" the proof of $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$ on the particular valuation v_0, and to compute on v_0 the function resulting from the interpretation of the proof. Thus our system and its constructive interpretation provide Prawitz's reductions for classical logic with a precise computational meaning. Besides, normalization can also be seen as a sort of *compactness* argument, producing a finite inconsistent valuation out of an infinite one.

Prawitz's usual natural deduction system for classical logic and his set of reduction rules will be recalled in Section 2. The witness extraction method devised in [1] will be described in Section 3. An example of witness extraction on a simple derivation will be given in Section 6. In Section 4, we shall present the *valuation interpretation* of classical natural deduction, that is, our system VS. In Subsection 4.4 we shall show how, in our valuations setting, each inference rule has a computational meaning. Then each of Prawitz's reduction rules will be shown in Subsection 4.5 in order to make explicit our interpretation of *classical* logical connectives and proofs. The extraction procedure of [1] will then be interpreted in the context of our valuation system in Subsection 4.6.[1]

2 Natural deduction for classical logic and Prawitz's reduction rules In this section we recall the system of Natural Deduction for Classical Logic [8] and the set of reduction rules devised by Prawitz.

2.1 Natural deduction for classical logic

$$\dfrac{\begin{array}{c}[\neg A]\\ \vdots \\ \bot\end{array}}{A} \qquad\qquad \dfrac{A \quad \neg A}{\bot}$$

$$\dfrac{A_1 \quad A_2}{A_1 \wedge A_2} \qquad\qquad \dfrac{A_1 \wedge A_2}{A_i}\ i = 1, 2$$

$$\dfrac{A}{\forall x.A}\,(*) \qquad\qquad \dfrac{\forall x.A}{A(t)}$$

(*) x not free in the assumptions on which A depends.

We have used $\neg$, $\wedge$ and $\forall$ as primitive logical connectives. All the other usual connectives can be defined in the usual way for classical logic.

$$A \vee B =_{Def} \neg(\neg A \wedge \neg B)$$

$$A \rightarrow B =_{Def} \neg(A \wedge \neg B)$$

$$\exists x.A =_{Def} \neg\forall x.\neg A$$

We can extend first-order classical logic by adding to it any Post system, that is, any set of atomic axioms and rules such as

$$\dfrac{}{Q} \qquad\qquad \dfrac{Q_1 \quad Q_2 \ \ldots\ Q_n}{Q_{n+1}}$$

where Q, Q_i are atomic formulas. When talking of witness extraction we will assume to be concerned with such extended systems. We now present Prawitz's reduction rules for classical logic [8] which we divide into two sets.

2.2 Prawitz's reductions

2.2.1 Reductions to eliminate detours

$$\dfrac{\dfrac{\begin{array}{c}[A]\\ \vdots \\ \bot\end{array}}{\neg A} \qquad \begin{array}{c}\vdots\\ A\end{array}}{\bot} \quad \rightsquigarrow \quad \begin{array}{c}\vdots\\ A\\ \vdots\\ \bot\end{array}$$

$$\dfrac{\dfrac{\begin{array}{cc}\vdots & \vdots\\ A_1 & A_2\end{array}}{A_1 \wedge A_2}}{A_i} \quad\rightsquigarrow\quad \begin{array}{c}\vdots\\ A_i\end{array} \qquad\qquad \dfrac{\dfrac{\begin{array}{c}\vdots\\ A\end{array}}{\forall x.A(x)}}{A(t)} \quad\rightsquigarrow\quad \begin{array}{c}\vdots\\ A(t)\end{array}$$

2.2.2 Reductions for double negation eliminations

$$\dfrac{\begin{array}{c}[\neg\neg A]\\ \vdots\\ \bot\end{array}}{\neg A} \quad\rightsquigarrow\quad \dfrac{\begin{array}{c}\dfrac{\dfrac{[\neg A] \qquad [A]}{\bot}}{\neg\neg A}\\ \vdots\\ \bot\end{array}}{\neg A}$$

$$\dfrac{\begin{array}{c}[\,\neg(A_1 \wedge A_2)\,]\\ \vdots\\ \bot\end{array}}{A_1 \wedge A_2} \quad\rightsquigarrow\quad \dfrac{\dfrac{\begin{array}{c}\dfrac{\dfrac{[\neg A_1] \qquad \dfrac{[A_1 \wedge A_2]}{A_1}}{\bot}}{\neg(A_1 \wedge A_2)}\\ \vdots\\ \bot\end{array}}{A_1} \qquad \dfrac{\begin{array}{c}\dfrac{\dfrac{[\neg A_2] \qquad \dfrac{[A_1 \wedge A_2]}{A_2}}{\bot}}{\neg(A_1 \wedge A_2)}\\ \vdots\\ \bot\end{array}}{A_2}}{A_1 \wedge A_2}$$

$$\dfrac{\begin{array}{c}[\neg\forall x.A(x)]\\ \vdots\\ \bot\end{array}}{\forall x.A(x)} \quad\rightsquigarrow\quad \dfrac{\dfrac{\begin{array}{c}\dfrac{\dfrac{[\neg A(y)] \qquad \dfrac{[\forall x.A(x)]}{A(y)}}{\bot}}{\neg\forall x.A(x)}\\ \vdots\\ \bot\end{array}}{A(y)}}{\forall y.A(y)}$$

It is easy to see that the reductions of the second set move the applications of the double negation elimination rule to simpler and simpler formulas.

Theorem 2.1 ([9][1]) *Proofs are strongly normalizable with respect to Prawitz's reductions.*

We can now consider one more reduction, called *trivial reduction*. Let Q be any atomic formula (possibly $\bot$), then

$$\begin{array}{c}\vdots\\ Q\\ \vdots\\ Q\end{array} \quad\rightsquigarrow\quad \begin{array}{c}\vdots\\ Q\end{array}$$

under the proviso that no assumption in the subproof of the inner Q is discharged in the whole proof. The above reduction rule is trivially strongly normalizing.

Lemma 2.2 ([1]) *Let us assume to have extended first-order classical logic with an inconsistent Post system (i.e., an inconsistent set of atomic axioms and rules). Let $\mathcal{D}$ be a derivation of $\perp$ which is closed and normal with respect to Prawitz's reduction rules and the trivial reduction. Then $\mathcal{D}$ contains only atomic axioms and rules.*

Prawitz used the normalizability of his set of rules in order to derive the consistency property of various systems. In the next section we shall see that, by means of the trivial reduction and Lemma 2.2, normalizability can be also used for computational purposes.

3 Witness extraction According to the BHK interpretation, an intuitionistic proof can provide constructive evidence for the formula it proves, by means of various extraction procedures. In a sense, then, a proof can be interpreted as being an *example* for the formula. This interpretation, as discussed in the introduction, is not possible for classical proofs. Kreisel, however, proposed for classical logic what he called *no-counterexample* interpretation. In it a classical proof is interpreted as the record of an unsuccessful attempt to describe a counterexample for the statement it proves. Even more, it can be seen as something that is able to refute any claimed counterexample (see, for instance, [10] 8.4). For Σ^0_1-formulas, *non*counterexamples coincide with examples (witnesses). So Kreisel's interpretation shows that it is possible to get witnesses out of classical proofs of Σ^0_1-formulas. This no-counterexample interpretation is exploited, as shown at the end of this section, in the witness extraction procedure proposed in [1] for classical proofs in natural deduction form.[2] This procedure will be outlined below and it makes essential use of the normalization property of Prawitz's reductions.

3.1 The witness extraction procedure of [1] Let us assume to have a closed derivation $\mathcal{D}$ of $\exists x.P(x)$ $(\neg\forall x.\neg P(x))$ where P is a decidable predicate. We first add the following (nonlogical) atomic rule to our system.

$$(r)\,\frac{P(x)}{\perp}$$

From that it is possible to get a closed proof for $\forall x.\neg P(x)$.

$$\cfrac{\cfrac{\cfrac{[P(x)]}{\perp}}{\neg P(x)}}{\forall x.\neg P(x)}$$

It is now possible to get a closed proof for $\perp$, as follows.

$$\cfrac{\begin{array}{c}\mathcal{D}\\ \neg\forall x.\neg P(x)\end{array} \qquad \cfrac{\cfrac{\cfrac{[P(x)]}{\perp}}{\neg P(x)}}{\forall x.\neg P(x)}}{\perp} \tag{2}$$

Such a derivation can be put in normal form by Theorem 2.1 and be further normalized with respect to the trivial reduction. Let $\mathcal{D}'$ be the derivation so obtained. By Lemma 2.2 we get that $\mathcal{D}'$ is formed only by atomic rules. Besides, it has necessarily to end with an application of the new nonlogical axiom (r) that we have added. So $\mathcal{D}'$ must have the form

$$\dfrac{\begin{matrix}\vdots\\ P(t)\end{matrix}}{\bot}$$

where t is now necessarily a witness for $\exists x.P(x)$, since $P(t)$ is derived only by using atomic axioms and rules different from $\frac{P(t)}{\bot}$, since the derivation is normal with respect to the trivial reduction.

We wish to stress that efficiency is not a concern of the above extraction procedure. It was not developed in order to have a procedure more efficient, for instance, than the "dumb" extraction algorithm (the algorithm that takes a classical proof of a Σ^0_1-sentence and runs through all the possible terms searching for a correct instance of the matrix of the theorem). The interest of the procedure described above lies instead in the use of Prawitz's reductions for computational purposes and in the fact that it exploits, in a precise way as shown below, Kreisel's no-counterexample interpretation.

3.2 Interpretation of the procedure as no-counterexample Kreisel's no-counterexample interpretation of classical logic looks at a classical proof as the evidence of the fact that no counterexample can be given for a proved statement. The above extraction procedure exploits such an interpretation for Σ^0_1-formulas. Adding to the system the rule (r) above amounts to claiming to have an argument showing the inconsistency, for any t, of assuming $P(t)$—it amounts to claiming to have a counterexample for $\exists x.P(x)$. The derivation (2) shows that if you have a counterexample for $\exists x.P(x)$, it *contradicts* what the derivation $\mathcal{D}$ asserts. The normalization procedure, however, *destroys* the claimed counterexample by providing a witness for our existentially quantified decidable predicate. The proof of $\exists x.P(x)$ behaves then, by means of the reduction rules, as a *noncounterexample*. In Section 6 we shall give an example of witness extraction for a proof in classical natural deduction which uses a simple Post system.

4 The valuation system for classical logic The extraction procedure outlined above cannot be of much help by itself in improving our understanding of the computational features of classical logic. In order really to be of help, it should make explicit some computational interpretation of classical formulas and proofs, in the same way in which the β-rule makes explicit the constructive meaning of the *intuitionistic* connective "$\rightarrow$" in the BHK interpretation. The aim of the rest of the paper then is to give a precise computational meaning to Prawitz's reductions in the context of an interpretation of classical logic. Instead of simply logical formulas we shall consider judgments on formulas and valuations. These will be given a computational interpretation that will be shown to be preserved by the logical rules of natural deduction. Moreover, we shall see how Prawitz's reduction rules make explicit such computational interpretation implicitly contained in derivations.

The bases of our system of judgments are the notions of assertion and valuation. In a sense this system, called Valuations System (VS), can be viewed as a formal system to reason about assertions and valuations.

4.1 Assertions An *assertion* (on A) is an expression of the form $\mathbf{t}.A$ or of the form $\mathbf{f}.A$, where A is a logical formula. In what follows, γ will be a variable ranging over the set $\{\mathbf{t}, \mathbf{f}\}$, while $\overline{\gamma}$ is defined in the following way.

$$\overline{\gamma} = \begin{cases} \mathbf{f} & \text{if } \gamma = \mathbf{t} \\ \mathbf{t} & \text{if } \gamma = \mathbf{f} \end{cases}$$

The intuitive meanings of the assertions $\mathbf{t}.A$ and $\mathbf{f}.A$ are "A is true" and "A is false," respectively. An assertion is in *canonical* form if it is not of the form $\mathbf{t}.\neg A$, $\mathbf{f}.\neg A$ and no double negation is present. We shall identify each formula with its equivalent canonical form, obtained by removing all double negations and replacing $\mathbf{t}.\neg A$, $\mathbf{f}.\neg A$ with $\mathbf{f}.A$ and $\mathbf{t}.A$, respectively. For instance, the canonical form of $\mathbf{t}.\neg\neg\neg(\neg\neg A_1 \wedge \neg\neg\neg A_2)$ is $\mathbf{f}.A_1 \wedge \neg A_2$.

4.2 Valuations Our system deals with classical validity of assertions in partial models (valuations) which can be seen as sets of assertions on atomic formulas. A *valuation* is a set of assertions on atomic formulas.

To formally deal with valuations we need a syntax for them. A *valuation-term* is an expression that denotes a valuation, and it is formed by using the following rules.

1. $\varnothing$ is a valuation-term (intuitively the "empty" valuation).
2. Valuation variables ($v, w, v_1, w_1, \ldots$) are valuation-terms.
3. If v and v' are valuation terms, then $\mathsf{v} + \mathsf{v}'$ is a valuation-term (intuitively the "union" of the valuations).
4. If $\{x_1, \ldots, x_m\} = FV(P)$ and P is atomic, then $\{\gamma.P \mid x_1, \ldots, x_m\}$ is a valuation-term (intuitively the valuation consisting in all possible *closed* instances of $\gamma.P$). If P is closed ($m = 0$) we shall use the notation $\{\gamma.P\}$.

From the syntax and the informal explanations given above, it is quite clear what is the valuation, or the set of valuations in case of a term with valuation variables, denoted by a valuation-term. In the following we shall not distinguish between valuation-terms and the valuations they denote. So, we can state $\mathsf{v} \subseteq \mathsf{v}'$, for v, v' valuation terms.

A valuation (-term) is *finite* if it consists of a finite number of closed atomic assertions. In particular, if P is open and the Herbrand universe of the logical language is infinite, $\{\gamma.P \mid x_1, \ldots, x_m\}$ is not a finite valuation (-term). The finiteness of a valuation term will be denoted by means of the subscript "*fin*."

If v is a finite valuation term, we shall denote by $\bigwedge \mathsf{v}$ the formula consisting in the conjunction of its assertions (considering $\mathbf{t}.P$ as P and $\mathbf{f}.P$ as $\neg P$). In the following we shall equate the valuation terms denoting the same valuation.

4.3 System *VS* System *VS* is a system about valuations, more precisely about assertions validated by partial models represented by valuations. We have defined above the terms of our valuation system: the valuation-terms. The well-formed formulas (judgments) of system *VS* are expressions of the following form,

$$\mathsf{v} \models \gamma.A$$

where v is a valuation term and $\gamma.A$ an assertion.

We now formally define the *intended constructive meaning* of our judgments. The rules of system *VS* are sound with respect to this intended meaning and the reduction rules will provide a means to make it explicit.

$$\begin{array}{lll}
\mathsf{v} \models \mathbf{t}.\bot & \Longleftrightarrow & \text{there exists } \mathsf{v}'_{fin} \subseteq \mathsf{v} \text{ s.t. } \bigwedge \mathsf{v}'_{fin} \to \bot \\
\mathsf{v} \models \mathbf{t}.P & \Longleftrightarrow & \text{there exists } \mathsf{v}'_{fin} \subseteq \mathsf{v} \text{ s.t. } \bigwedge \mathsf{v}'_{fin} \to P \\
 & & \text{for } P \text{ atomic} \\
\mathsf{v} \models \mathbf{t}.\forall x.A(x) & \Longleftrightarrow & \text{for any term } t,\ \mathsf{v} \models \mathbf{t}.A(t) \\
\mathsf{v} \models \mathbf{t}.A \wedge B & \Longleftrightarrow & \mathsf{v} \models \mathbf{t}.A \text{ and } \mathsf{v} \models \mathbf{t}.B \\
\mathsf{v} \models \mathbf{f}.A & \Longleftrightarrow & \text{for any } \mathsf{v}' \text{ s.t. } \mathsf{v}' \models \mathbf{t}.A,\ \ \mathsf{v}+\mathsf{v}' \models \mathbf{t}.\bot,
\end{array}$$

where $\bigwedge \mathsf{v}'_{fin}$ is the logical conjunction of all the formulas corresponding to the assertions in v'_{fin}, and $\bigwedge \mathsf{v}'_{fin} \to \bot$ and $\bigwedge \mathsf{v}'_{fin} \to P$ must be classically valid formulas. Then a formula $\mathsf{v} \models \gamma.A$ has an interpretation similar to Kripke's forcing semantics.

We will use the notation '$\mathbf{inc}(\mathsf{v})$' as short for '$\mathsf{v} \models \mathbf{t}.\bot$'. The rules of our system are the following.

Rules for valuations

$$\frac{\mathsf{v} \models \gamma.A}{\mathsf{v}+\mathsf{v}' \models \gamma.A}\ \text{for any } \mathsf{v}'$$

$$\frac{}{\{\gamma.P \mid x_1, \dots x_n\} \models \gamma.P(t_1, \dots, t_n)}\ \text{for any } t_1, \dots, t_n$$

Logical rules

$$\frac{\begin{array}{c}[v \models \overline{\gamma}.A] \\ \vdots \\ \mathbf{inc}(v+\mathsf{v})\end{array}}{\mathsf{v} \models \gamma.A}\ (*) \qquad \frac{\mathsf{v} \models \gamma.A \quad \mathsf{v}' \models \overline{\gamma}.A}{\mathbf{inc}(\mathsf{v}+\mathsf{v}')}$$

$$\frac{\mathsf{v}_1 \models \mathbf{t}.A_1 \quad \mathsf{v}_2 \models \mathbf{t}.A_2}{\mathsf{v}_1+\mathsf{v}_2 \models \mathbf{t}.A_1 \wedge A_2} \qquad \frac{\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2}{\mathsf{v} \models \mathbf{t}.A_i}\ i = 1, 2$$

$$\frac{\mathsf{v} \models \mathbf{t}.A}{\mathsf{v} \models \mathbf{t}.\forall x.A}\ (**) \qquad \frac{\mathsf{v} \models \mathbf{t}.\forall x.A}{\mathsf{v} \models \mathbf{t}.A(u)}$$

(*) It is possible to discharge only assumptions with valuation-terms consisting of a single variable; besides, the valuation-variable v cannot occur free in the assumptions on which $\mathbf{inc}(v + \mathsf{v})$ depends, but $v \models \overline{\gamma}.A$.

(**) x is not free in the assertions of the assumptions on which $\mathsf{v} \models \mathbf{t}.A$ depends.

In this system if we have two equal valuation-terms or two equal assertions we can always substitute one for the other.

In the case in which we consider extensions of natural deduction with Post systems, we must extend system VS as follows.

$$\frac{}{\varnothing \models \mathbf{t}.Q} \qquad \frac{v_1 \models \mathbf{t}.Q_1 \ \ldots \ v_n \models \mathbf{t}.Q_n}{\sum_{i=1}^{n} v_i \models \mathbf{t}.Q_{n+1}}$$

Remark 4.1 We have given rules for conjunction and universal quantification only for the assertions beginning with $\mathbf{t}$. The rules dealing with conjunction and universal quantification in the other case, namely

$$\frac{\mathsf{v} \models \mathbf{f}.A_i}{\mathsf{v} \models \mathbf{f}.A_1 \wedge A_2}\ i = 1, 2 \qquad \frac{\mathsf{v} \models \mathbf{f}.A}{\mathsf{v} \models \mathbf{f}.\forall x.A}$$

are redundant. They correspond to rules for disjunction and existential quantification and, since we are in a classical context, they are derivable.

$$\dfrac{\dfrac{\dfrac{[w \models \mathbf{t}.A_1 \wedge A_2]}{w \models \mathbf{t}.A_i} \qquad \mathsf{v} \models \mathbf{f}.A_i}{\mathbf{inc}(w + \mathsf{v})}}{\mathsf{v} \models \mathbf{f}.A_1 \wedge A_2} \qquad \dfrac{\dfrac{\dfrac{[w \models \mathbf{t}.\forall m.A]}{w \models \mathbf{t}.A} \qquad \mathsf{v} \models \mathbf{f}.A}{\mathbf{inc}(w + \mathsf{v})}}{\mathsf{v} \models \mathbf{f}.\forall m.A}$$

In the following we shall use the first two rules of the above remark as abbreviations for their derivations.

4.4 Interpretation of the deduction rules The logical rules of system VS clearly are an elaboration of those of Prawitz's natural deduction for classical logic, recalled in Section 2. The rules for valuations, instead, do not have any counterpart in Prawitz's system and have been introduced since they are needed if you wish to deal with valuations. Both of the above groups of rules, however, have a precise interpretation in terms of the intended meaning of the formulas in our valuation system. Furthermore the rules *effectively* show how to get the intended meaning of the conclusion if we have one of the premises. For the axiom

$$\frac{}{\{\mathbf{t}.P \mid x_1, \ldots x_n\} \models \mathbf{t}.P(t_1, \ldots, t_n)}$$

it is easy to see that if $t_1, \ldots, t_n$ are closed then $\{\mathbf{t}.P(t_1, \ldots, t_n)\}$ is the finite valuation which satisfies the formula on the right. The most interesting rule is obviously

$$\dfrac{\begin{matrix}[v \models \overline{\gamma}.A] \\ \vdots \\ \mathbf{inc}(v + \mathsf{v})\end{matrix}}{\mathsf{v} \models \gamma.A}$$

It is indeed two different rules, according to what γ actually is. If $\gamma \equiv \mathbf{t}$, it corresponds to the introduction of the negation and expresses exactly the intended meaning of $\mathsf{v} \models \gamma.A$. If $\gamma \equiv \mathbf{f}$, instead, the rule corresponds to the elimination of double negation and it should justify the interpretation of $\mathsf{v} \models \mathbf{t}.A$. If A is an atomic formula P then the expected meaning of $\mathsf{v} \models \mathbf{t}.A$ is that there exists a finite subvaluation of v from which P is derivable. To see how the rule justifies such an interpretation, let us note first that the derivation above the premise of the rule is a derivation which allows us to infer $\mathbf{inc}(\mathsf{v} + \mathsf{v}')$ once we have proved that $\mathsf{v}' \models \mathbf{f}.P$. The meaning of $\mathbf{inc}(\mathsf{v} + \mathsf{v}')$ is that there exists a finite valuation of $\mathsf{v} + \mathsf{v}'$, from which it is possible to derive the falsehood. Then, from the finite valuation we could get from $\mathbf{inc}(\mathsf{v} + \mathsf{v}')$, it would be possible to get the finite valuation for $\mathsf{v} \models \mathbf{t}.P$. The problem now is to find a v' and a proof of $\mathsf{v}' \models \mathbf{f}.P$. By using our rule for valuations it is immediate to get $\mathsf{v}' \models \mathbf{f}.P$ if we take $\mathsf{v}' \equiv \{\mathbf{f}.P \mid x_1, \ldots, x_n\}$. So the interpretation of $\mathsf{v} \models \mathbf{t}.A$ seems to be justified by the rule. There is however a case yet to be considered in the argument above: A could be a nonatomic formula. This, however, is not a difficult obstacle to overcome since we can always transform a deduction in such a way that double negation elimination is performed only on atomic formulas. This sort of transformation is exactly the one performed by Prawitz's reductions, which then manage to make explicit the constructive contents of the elimination of double negation. Later on we shall deal with the interpretation of Prawitz's reductions.

As to what concerns the rules for introducing and eliminating $\wedge$ and $\forall$, it is clear how they make evident the intended meaning of the formulas in their conclusions.

Finally, the rule

$$\frac{\mathsf{v} \models \gamma.A \qquad \mathsf{v}' \models \overline{\gamma}.A}{\mathbf{inc}(\mathsf{v} + \mathsf{v}')}$$

expresses the condition for a valuation to be inconsistent. It corresponds to the rule of modus ponens in intuitionistic logic, which has a (BHK) interpretation in terms of application of a function to its argument. According to our interpretation, by the fact that $\gamma.A$ is equivalent to $\overline{\gamma}.\neg A$, it can be interpreted as two different applications according to which one of the two subproofs is seen as the function. Alternatively, it can be seen as a *symmetric* form of application.

Now we state the almost immediate formal equivalence between our valuation calculus *VS* and classical natural deduction.

Theorem 4.2 (Validity Theorem) *If a formula D is derivable in classical logic from assumptions $G_1, \ldots, G_n$ then there exists a derivation in VS of $\sum_{j=1}^{n} w_j \models \mathbf{t}.D$ from assumptions $w_1 \models \mathbf{t}.G_1, \ldots, w_n \models \mathbf{t}.G_n$ where $w_1, \ldots, w_n$ are valuation variables.*

Proof: By induction on derivations. $\square$

4.5 Reduction rules for system *VS* It is quite straightforward to extend Prawitz's reduction rules for classical natural deduction to system *VS*. The former ones, as briefly discussed in Subsection 3, enable us to extract constructive contents from proofs of Σ^0_1-sentences. We have seen, in the previous subsection, that the double negation elimination rule has no explicit constructive meaning when applied to nonatomic formulas. Prawitz's reductions for double negation elimination enable us to transform a proof in such a way that such rules are applied only to atomic formulas, and hence to get proofs containing deduction rules all of which have explicit constructive meaning, with respect to our interpretation of judgments.

What is stated above gives a constructive sense to the normalization process as a whole. What is still lacking is a computational interpretation of the single steps of the process, that is, of the single reduction rules. Then we need to show how each reduction rule makes more and more explicit our interpretation of the *classical* logical connectives in the partial models interpretation.

Prawitz's reductions are divided into two groups: those eliminating detours and those simplifying the structure of formulas to which the double negation elimination rule is applied. The former ones also maintain in the *VS* setting their meaning of "simplification." In fact, we have seen that the rule

$$\frac{\mathsf{v} \models \overline{\gamma}.A \qquad \mathsf{v}' \models \gamma.A}{\mathbf{inc}(\mathsf{v}+\mathsf{v}')}$$

is a sort of modus ponens. Then if $\mathsf{v}' \models \gamma.A$ has been obtained by means of the rule

$$\frac{\begin{array}{c}[v \models \gamma.A]\\ \vdots \\ \mathbf{inc}(v+\mathsf{v})\end{array}}{\mathsf{v} \models \overline{\gamma}.A}$$

we can apply the reduction

$$\frac{\dfrac{\begin{array}{c}[v \models \gamma.A]\\ \vdots \\ \mathbf{inc}(v+\mathsf{v})\end{array}}{\mathsf{v} \models \overline{\gamma}.A} \qquad \begin{array}{c}\vdots\\ \mathsf{v}' \models \gamma.A\end{array}}{\mathbf{inc}(\mathsf{v}+\mathsf{v}')} \quad \rightsquigarrow \quad \begin{array}{c}\vdots\\ \mathsf{v}' \models \gamma.A\\ \vdots\\ \hline \mathbf{inc}(\mathsf{v}+\mathsf{v}')\end{array}$$

which has a meaning similar to the β-rule for the λ-calculus.

If we know that (the interpretation of) $\mathsf{v}_1+\mathsf{v}_2 \models \mathbf{t}.A_i$ holds because we have inferred it from the fact that $\mathsf{v}_1 \models \mathbf{t}.A_1$ *and* $\mathsf{v}_2 \models \mathbf{t}.A_2$ hold, then we can avoid passing through $\mathsf{v}_1+\mathsf{v}_2 \models \mathbf{t}.A_1 \wedge A_2$ simply by extending the valuation v_i to $\mathsf{v}_1+\mathsf{v}_2$. This justifies the following reduction.

$$\frac{\dfrac{\begin{array}{c}\vdots\\ \mathsf{v}_1 \models \mathbf{t}.A_1\end{array} \qquad \begin{array}{c}\vdots\\ \mathsf{v}_2 \models \mathbf{t}.A_2\end{array}}{\mathsf{v}_1+\mathsf{v}_2 \models \mathbf{t}.A_1 \wedge A_2}}{\mathsf{v}_1+\mathsf{v}_2 \models \mathbf{t}.A_i} \quad \rightsquigarrow \quad \frac{\begin{array}{c}\vdots\\ \mathsf{v}_i \models \mathbf{t}.A_i\end{array}}{\mathsf{v}_1+\mathsf{v}_2 \models \mathbf{t}.A_i}$$

In a way similar to the previous one, a justification can also be given for the rule

$$\cfrac{\cfrac{\mathsf{v} \models \mathbf{t}.A}{\mathsf{v} \models \mathbf{t}.\forall x.A}}{\mathsf{v} \models \mathbf{t}.A(t)} \quad \leadsto \quad \begin{array}{c}\vdots \\ \mathsf{v} \models \mathbf{t}.A(t)\end{array}$$

The rules peculiar to classical logic, and to which no computational interpretation has ever been given, are the ones dealing with double negation elimination. A double negation elimination rule can be seen as a process to get the intended meaning of the conclusion only in the case in which its formula is atomic. Then, in a proof if the conclusion of an application of the double negation elimination rule is a compound formula, we can "decompose" it—in the case in which it is a conjunction, using the following rule.

$$\cfrac{\begin{array}{c}[v \models \mathbf{f}.A_1 \wedge A_2] \\ \vdots \\ \mathbf{inc}(v + \mathsf{v})\end{array}}{\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2} \quad \leadsto \quad \cfrac{\cfrac{\begin{array}{c}\cfrac{v \models \mathbf{f}.A_1}{v \models \mathbf{f}.A_1 \wedge A_2} \\ \vdots \\ \mathbf{inc}(v + \mathsf{v})\end{array}}{\mathsf{v} \models \mathbf{t}.A_1} \qquad \cfrac{\begin{array}{c}\cfrac{v \models \mathbf{f}.A_2}{v \models \mathbf{f}.A_1 \wedge A_2} \\ \vdots \\ \mathbf{inc}(v + \mathsf{v})\end{array}}{\mathsf{v} \models \mathbf{t}.A_2}}{\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2}$$

The intended meaning of $\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2$, that is, a pair formed by the meaning of $\mathsf{v} \models \mathbf{t}.A_1$ *and* the meaning of $\mathsf{v} \models \mathbf{t}.A_2$ (see 4.3), is explicit only in the case the judgment $\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2$ has been obtained by means of an application of the $\wedge$-introduction rule. In the case we are considering, instead, $\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2$ has been obtained by means of a double negation elimination. Its meaning is then "hidden," and must be made explicit.

The reduction rule above, in fact, makes $\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2$ derived from two derivations, one for $\mathsf{v} \models \mathbf{t}.A_1$ and one for $\mathsf{v} \models \mathbf{t}.A_2$, thus enabling us to interpret the derivation of $\mathsf{v} \models \mathbf{t}.A_1 \wedge A_2$ as a pair. Of course, the pair obtained is a pair of constructive meanings only in the case in which the conjuncts are atomic. Otherwise we must perform other reduction steps to get to their atomic components.

An argument similar to the previous one can explain the computational meaning of Prawitz's reduction rule for double negation elimination on universally quantified formulas. This reduction has the following form in the context of system *VS*.

$$\cfrac{\begin{array}{c}[v \models \mathbf{f}.\forall m.A] \\ \vdots \\ \mathbf{inc}(v + \mathsf{v})\end{array}}{\mathsf{v} \models \mathbf{t}.\forall m.A} \quad \leadsto \quad \cfrac{\cfrac{\begin{array}{c}\cfrac{[v \models \mathbf{f}.A]}{v \models \mathbf{f}.\forall m.A} \\ \vdots \\ \mathbf{inc}(v + \mathsf{v})\end{array}}{\mathsf{v} \models \mathbf{t}.A}}{\mathsf{v} \models \mathbf{t}.\forall m.A}$$

4.6 The witness extraction procedure in the** VS **setting It is possible to apply the extracting procedure of [1], recalled in Section 3, directly in system *VS*. (An example can easily be obtained by rephrasing the one in Section 6, as explained at the end of it.) Using system VS and the interpretation of its expressions, rules and reductions, the meaning of what the procedure does is made clearer.

Let us assume to have a closed proof in classical logic of $\exists x.P(x)(\equiv \neg\forall x.\neg P(x))$. By the Validity Theorem it is possible to get a derivation of $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$ in system VS. According to our interpretation, the proof of this statement can be viewed as a function that, whenever applied to a valuation v_0 validating $\mathbf{t}.\forall x.\neg P(x)$, returns a finite subvaluation of v_0 from which falsehood can be derived, that is, an inconsistent one. It is clear that if $\mathsf{v}_0 \equiv \{\mathbf{f}.P \mid x\}$ then we would also have the possibility of getting a witness for the statement $\exists x.P(x)$, since in the finite subvaluation $\{\mathbf{f}.P(t_1),\ldots \mathbf{f}.P(t_n)\}$ we would get from v_0, one of the t_i's is necessarily a witness.

The problem now is how to "feed" $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$ on the valuation $\{\mathbf{f}.P \mid x\}$. This could be done in the following way, in the case in which we have a derivation $\mathcal{D}$ for $\{\mathbf{f}.P \mid x\} \models \mathbf{t}.\forall x.\neg P(x)$.

$$\frac{\begin{array}{c}\vdots\\ \varnothing \models \mathbf{f}.\forall x.\neg P(x)\end{array} \qquad \begin{array}{c}\mathcal{D}\\ \{\mathbf{f}.P \mid x\} \models \mathbf{t}.\forall x.\neg P(x)\end{array}}{\mathbf{inc}(\{\mathbf{f}.P \mid x\})}$$

It is easy to see, however, that this is not possible since it would imply that it is possible to obtain $\forall x.\neg P(x)$ from a finite subvaluation of $\{\mathbf{f}.P \mid x\}$. We had a similar problem in Section 3 where, by interpreting the proof of $\exists x.P(x)$ as a counterexample destructor, we found it necessary to "feed" it on a counterexample. We decided there to add the rule

$$(r)\frac{P(x)}{\bot},$$

which can be interpreted as the claim of having a counterexample.

Here we can provide the derivation of $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$, interpreted as a function, with the valuation $\{\mathbf{f}.P \mid x\}$, by adding a rule to system VS which corresponds to the rule (r) above, that is, the following one.

$$(r_{VS})\frac{v \models \mathbf{t}.P(x)}{\mathbf{inc}(v + \{\mathbf{f}.P \mid x\})}.$$

By following our interpretation for VS expressions, such a rule says that it is possible to get a finite inconsistent valuation out of the valuation $\mathsf{v} + \{\mathbf{f}.P \mid x\}$ for any valuation v validating $\mathbf{t}.P(x)$. The *application* of $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$ to the valuation $\{\mathbf{f}.P \mid x\}$ can now be performed in VS as follows.

$$\frac{\begin{array}{c}\vdots\\ \varnothing \models \mathbf{f}.\forall x.\neg P(x)\end{array} \qquad \dfrac{\dfrac{\dfrac{[v \models \mathbf{t}.P(x)]}{\mathbf{inc}(v + \{\mathbf{f}.P \mid x\})}}{\{\mathbf{f}.P \mid x\} \models \mathbf{t}.\neg P(x)}}{\{\mathbf{f}.P \mid x\} \models \mathbf{t}.\forall x.\neg P(x)}}{\mathbf{inc}(\{\mathbf{f}.P \mid x\})}$$

In Section 3, by means of rule (r), we managed to get $\bot$, that is, *only* the information that something inconsistent had been added to the system. We knew, from *outside* the system, that such an inconsistency depended on rule (r) and that necessarily there had to be some inconsistent $P(t_1), \ldots, P(t_n)$. System VS, instead, is more informative than the bare natural deduction. Here formulas have a constructive interpretation, as seen in previous sections. In particular, in the present case we have managed to derive more precise information than simply the presence of an inconsistency: we have derived $\mathbf{inc}(\{\mathbf{f}.P \mid x\})$, that is, the information that a finite subset of $\{\mathbf{f}.P \mid x\}$ is inconsistent. The information that there are some inconsistent $P(t_1), \ldots, P(t_n)$ has now been *internalized*. The reduction rules of the system enable us to get explicitly a finite subset $\{\mathbf{f}.P \mid x\}$. In fact, Lemma 2.2 holds clearly for system VS as well, if rules for valuations are not used in a derivation. So, a normal form of the above derivation is necessarily of the shape

$$(r_{VS})\,\frac{\begin{array}{c}\vdots\\ \varnothing \models \mathbf{t}.P(t)\end{array}}{\mathbf{inc}(\{\mathbf{f}.P \mid x\})}$$

where $\begin{array}{c}\vdots\\ \varnothing \models \mathbf{t}.P(t)\end{array}$ is a derivation made of atomic axioms and rules only. So $\{\mathbf{f}.P(t)\}$ is an inconsistent finite subvaluation of $\{\mathbf{f}.P \mid x\}$ and then the term t is a witness of $\exists x.P(x)$, since, by Lemma 2.2, $\varnothing \models \mathbf{t}.P(t)$ is obtained using only atomic axioms and atomic rules.

Let us note that in system VS we did not need introducing rule (r_{VS}), since such a rule is derivable, as shown by the deduction below.

$$\frac{v \models \mathbf{t}.P(x) \qquad \overline{\{\mathbf{f}.P \mid x\} \models \mathbf{f}.P(x)}}{\mathbf{inc}(v + \{\mathbf{f}.P \mid x\})}$$

We made the choice of adding rule (r_{VS}), however, in order to be able also to profit from Lemma 2.2 for system VS. The derivability of rule (r_{VS}) in VS is not in contradiction with the fact that rule (r), which we had added to the system of natural deduction in Section 3, is *obviously* not derivable in the latter system. Rule (r_{VS}) is weaker than (r). In fact, by means of it we managed to derive $\mathbf{inc}(\{\mathbf{f}.P \mid x\})$, a thing more informative and hence weaker, than the $\bot$ of bare natural deduction.

5 Conclusions We have developed a formal system VS for classical logic, whose well-formed formulas are *judgments* of the form $\mathsf{v} \models \gamma.A$, informally stating that the *partial model* denoted by the term v *validates* the *assertion* $\gamma.A$, where γ is $\mathbf{t}$ or $\mathbf{f}$. A precise constructive interpretation has been given to such judgments, depending on the main connective of A. Such an interpretation is respected by the deduction rules of the system. Moreover, a derivation which is normal with respect to Prawitz's reductions for classical logic can be interpreted as an effective evidence for the interpretation of the judgment it proves. Then Prawitz's reduction rules for classical logic turn out to be given a precise computational meaning, by means of our interpretation of proofs and judgments. This computational meaning was only implicit in the procedure of witness extraction from classical proofs of Σ_1^0-formulas devised in [1]. Finally, the witness extraction procedure of [1] has been viewed from the perspective of our interpretation.

6 *Appendix (A simple example of use of Prawitz's reductions)* Let us assume to extend first order classical logic with a Post system containing the axiom $\overline{P(0)}$. If we consider "$\neq$" as an atomic predicate, $P(0)$ could be, for instance, the formula $0 \neq 1$ as it is used in Peano's axiomatization of classical arithmetic. Let us now consider the following weird proof of the statement $\exists x.P(x)$ ($\equiv \neg\forall x.\neg P(x)$).

$$\dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(1)} \quad \dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(0)} \quad \dfrac{\dfrac{\dfrac{[\neg(\neg P(1) \wedge P(0))] \quad \dfrac{[\neg P(1)] \quad \overline{P(0)}}{\neg P(1) \wedge P(0)}}{\bot}}{\neg P(1) \wedge P(0)}}{P(0)}}{\bot}}{P(1)}}{\bot}}{\neg\forall x.\neg P(x)}$$

Let us denote by $\mathcal{D}$ the above derivation, and see now how our extraction procedure enables us to get a witness for $\exists x.P(x)$ from $\mathcal{D}$. We begin by adding the atomic rule

$$\dfrac{P(x)}{\bot}$$

to our system, and derive $\bot$ from it and $\mathcal{D}$.

$$\dfrac{\begin{array}{c}\mathcal{D} \\ \neg\forall x.\neg P(x)\end{array} \qquad \dfrac{\dfrac{\dfrac{[P(x)]}{\bot}}{\neg P(x)}}{\forall x.\neg P(x)}}{\bot}$$

We now proceed to normalize the above proof according to our reduction rules. To save space we begin by normalizing the subderivation $\mathcal{D}$. In $\mathcal{D}$ we have double negation elimination applied to $\neg(\neg P(1) \wedge P(0))$. We can use the reduction which modifies the deduction in such a way that the double negation elimination rule is applied to simpler formulas, namely $\neg\neg P(1)$ and $\neg P(0)$.

$$\dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(1)} \qquad \dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(0)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\neg\neg P(1)] \quad \dfrac{[\neg P(1) \wedge P(0)]}{P(1)}}{\bot}}{\neg(\neg P(1) \wedge P(0))} \quad \dfrac{[\neg P(1)] \quad \overline{P(0)}}{\neg P(1) \wedge P(0)}}{\bot}}{\neg P(1)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{[\neg P(0)] \quad \dfrac{[\neg P(1) \wedge P(0)]}{P(0)}}{\bot}}{\neg(\neg P(1) \wedge P(0))} \quad \dfrac{[\neg P(1) \quad \overline{P(0)}}{\neg P(1) \wedge P(0)}}{\bot}}{P(0)}}{\neg P(1) \wedge P(0)}}{P(0)}}{\bot}}{P(1)}}{\bot}}{\neg\forall x.\neg P(x)}$$

We can now eliminate the "detour" for the conjunction.

$$\dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(1)} \qquad \dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(0)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{[\neg P(0)] \quad \dfrac{[\neg P(1) \wedge P(0)]}{P(0)}}{\bot}}{\neg(\neg P(1) \wedge P(0))} \quad \dfrac{[\neg P(1)] \quad \overline{P(0)}}{\neg P(1) \wedge P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}}{\neg\forall x.\neg P(x)}$$

Next we apply the reduction for the detour corresponding to the introduction of the negation $\neg(\neg P(1) \wedge P(0))$ and its immediate elimination.

$$\dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(1)} \qquad \dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(0)} \qquad \dfrac{\dfrac{[\neg P(0)] \quad \dfrac{\dfrac{[\neg P(1)] \quad \overline{P(0)}}{\neg P(1) \wedge P(0)}}{P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}}{\neg\forall x.\neg P(x)}$$

Again we can apply the reduction to eliminate the detour for the conjunction. The resulting derivation is the normal form, with respect to Prawitz's reduction rules, of the subderivation $\mathcal{D}$. The complete derivation then has the following shape and is not in normal form.

$$\dfrac{\dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(1)} \quad \dfrac{\dfrac{\dfrac{[\forall x.\neg P(x)]}{\neg P(0)} \quad \dfrac{\dfrac{[\neg P(0)] \quad \overline{P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}}{\neg\forall x.\neg P(x)} \qquad \dfrac{\dfrac{\dfrac{[P(x)]}{\bot}}{\neg P(x)}}{\forall x.\neg P(x)}}{\bot}$$

In the above derivation, the negation $\neg\forall x.\neg P(x)$ is introduced and the rule for its elimination is applied. To eliminate this detour we apply the corresponding reduction.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[P(x)]}{\bot}}{\neg P(x)}}{\forall x.\neg P(x)}}{\neg P(1)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[P(x)]}{\bot}}{\neg P(x)}}{\forall x.\neg P(x)}}{\neg P(0)} \quad \dfrac{\dfrac{[\neg P(0)] \quad \overline{P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}$$

There are now two detours for the universal quantification to eliminate.

$$\dfrac{\dfrac{\dfrac{[P(1)]}{\bot}}{\neg P(1)} \qquad \dfrac{\dfrac{\dfrac{\dfrac{[P(0)]}{\bot}}{\neg P(0)} \quad \dfrac{\dfrac{[\neg P(0)] \quad \overline{P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}$$

Let us first eliminate the leftmost detour for the negation.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[P(0)]}{\bot}}{\neg P(0)} \quad \dfrac{\dfrac{[\neg P(0)] \quad \overline{P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}$$

We have still one more detour for the negation to eliminate.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\neg P(0)] \quad \overline{P(0)}}{\bot}}{P(0)}}{\bot}}{P(1)}}{\bot}$$

At last we have a deduction which is normal with respect to Prawitz's reductions. We can now apply the trivial reduction:

$$\cfrac{\cfrac{\cfrac{\overline{P(0)}}{\bot}}{P(1)}}{\bot}$$

One more trivial reduction can be applied, getting a deduction in normal form with respect to both Prawitz's reductions and the trivial one:

$$\cfrac{\overline{P(0)}}{\bot}$$

The witness we needed then is 0. Due to the simplicity of our example, the derivation for $P(0)$ consists in an axiom. If we had a more complex Post system we could have more complex atomic derivations for the atomic predicate displaying the witness.

It is easy to see that we could use the example presented above in the setting of system VS. In such a case our initial proof of $\neg\forall x.\neg P(x)$ could easily be elaborated in order to get a proof in VS of $\varnothing \models \mathbf{f}.\forall x.\neg P(x)$. By adding to the system the rule (r_{VS}), as described in Subsection 4.6, we manage to obtain, in a way similar to getting $\bot$ above, a derivation for $\mathbf{inc}(\{\mathbf{f}.P \mid x\})$ which, according to our interpretation, can be seen as a finite inconsistent subvaluation of $\{\mathbf{f}.P \mid x\}$. This inconsistent subvaluation, however, is only implicitly represented by the derivation. We can explicitly obtain it only by means of a normalization process. If we follow the same reduction steps described above, it is possible to obtain the derivation

$$\cfrac{\overline{\varnothing \models \mathbf{t}.P(0)}}{\mathbf{inc}(\{\mathbf{f}.P \mid x\})}$$

which explicitly takes to the finite inconsistent subvaluation $\{\mathbf{f}.P(0)\} \subseteq \{\mathbf{f}.P \mid x\}$.

Acknowledgments We would like to thank Professor Mariangiola Dezani for her guidance and Elena Barbanera for her help to improve the style of the presentation of the paper. The first author also wishes to express his gratitude to Sabrina Agosto for her invaluable support.

NOTES

1. An incomplete and preliminary version of the present paper was presented in [2].
2. Indeed in [1] a stronger result is proved, namely that it is possible to get a witness from proofs of Σ^0_1-sentences in Higher-Order Classical Logic.

REFERENCES

[1] Barbanera F., and S. Berardi, "Witness Extraction in Classical Logic through Normalization," pp. 219–246 in *Logical Environments*, edited by G. Huet and G. Plotkin, Cambridge University Press, Cambridge, 1993. MR 1255117 1, 1, 1, 1, 1, 1, 2.1, 2.2, 3, 3.1, 4.6, 5, 5, 6

[2] Barbanera, F., and S. Berardi, “A constructive valuation interpretation for classical logic and its use in witness extraction,” pp. 1–23 in *Proceedings of Colloquium on Trees in Algebra and Programming (CAAP)*, LNCS 581, Springer-Verlag, New York, 1992. MR 94h:03115 6

[3] Friedman, H., “Classically and intuitionistically provably recursive functions,” pp. 21–28 in *Higher Set Theory*, edited by D. S. Scott and G. H. Muller, Lecture Notes in Mathematics, vol. 699, Springer-Verlag, New York, 1978. Zbl 0396.03045 MR 80b:03093 1

[4] Heyting, A., *Mathematische Grundlagenforschung. Intuitionismus. Beweistheorie*, Springer, Berlin, Reprinted 1974. Zbl 0278.02002 MR 49:8806 1

[5] Kreisel, G., “Mathematical significance of consistency proofs,” *The Journal of Symbolic Logic*, vol. 23 (1958), pp. 155–182. Zbl 0088.01502 MR 22:6710 1

[6] Kolmogorov, A. N., “Zur Deutung der Intuitionistischen Logik,” *Mathematische Zeitschrift*, vol. 35 (1932), pp. 58–56. Zbl 0004.00201 1

[7] Martin-Löf, P., “An intuitionistic theory of types: predicative part,” pp. 73–118 in *Logic Colloquium 73*, edited by H. E. Rose and J. C. Sheperdson, North-Holland, Amsterdam, 1975. Zbl 0334.02016 MR 52:7856 1

[8] Prawitz, D., *Natural deduction, a proof theoretical study*, Almqvist and Winskell, Stockholm, 1965. Zbl 0173.00205 MR 33:1227 1, 1, 1, 2, 2.1

[9] Prawitz, D., “Validity and normalizability of proofs in first and second order classical and intuitionistic logic,” pp. 11–36 in *Atti del I Congresso Italiano di Logica*, Bibliopolis, Napoli, 1981. 1, 2.1

[10] Shoenfield, J. R., *Mathematical Logic*, Addison-Wesley, Reading, 1967. Zbl 0155.01102 MR 37:1224 3

Universitá di Torino
Dipartimento di Informatica
Corso Svizzera, 185
I-10149 Torino
ITALY
email: barba@di.unito.it
email: stefano@di.unito.it