The Collection Method in Second-Order Intuitionistic Logic

Marco Benini

Dipartimento di Scienze dell'Informazione
via Comelico 39, Milano, Italy
beninim@dsi.unimi.it

Abstract. The collection method is a procedural way to extract information from proofs in first-order logics. In this paper we extend the collection method to second-order intuitionistic logic, and we derive its relevant properties. We will show constructiveness of second-order intuitionistic logic, giving an optimal proof, since we use induction limited to the ω ordinal. Our results can be immediately extended to any finite-order intuitionistic logic, both predicate and propositional, and to many important theories, such as Heyting arithmetic.

1 Introduction

It is common experience that the proof of a theorem is more informative than just its statement. We are not interested in a philosophical view – a deduction gives evidence to propositions – but in the technical question: how can we extract information from a proof?

If we have a formal deduction in a given logical system, for example, natural calculus for first-order logic, how can we make explicit the knowledge involved in the proof? What kind of information are we able to extract? To what extent?

An answer to these questions could be given only if we define in a precise way what we intend with the word "information". We adopt the simplest (and safest) definition: the information contained in a proof is the set of facts that are used to prove the statement. The Normalization Theorem gives us an instrument to generate this set. We take the proof in normal form, and we scan it from the root, collecting conclusions of subproofs.

This method is unsatisfactory for several reasons:

- There are interesting logics and theories for which no normalization procedure exists.
- While it doesn't introduce new information, this process can destroy the existing one.
- Its properties (such as induction depth) strongly depend on details of calculi, such as the syntax of terms and formulas.

Its main advantage lies in the possibility of proving the consistency of the underlying calculus or theory, along with its constructivity.

These disadvantages of the normalization procedure are not surprising, since constructivity is a byproduct of the theorem. The Normalization Theorem was conceived [12] as a syntactical way to prove consistency. It is the principal instrument to analyze the role of proofs in logic, and most results in proof theory are related to normalization, or cut-elimination, if you use a sequent-like or tableaux-like calculus.

But when we focus on constructivity, the Normalization Theorem is not the right instrument. It is too powerful, so it does not hold in many calculi and logics, and it is too general to produce optimal results. As stated before, we are interested in extracting information from proofs, and, in this view, a normalization procedure is not a good solution [13].

Another way to extract information from a formal proof (in first-order logics) is the collection method [16, 17]. Its main features are:

- It can be applied to several first-order logics and theories for which the Normalization Theorem fails.
- It does not add new information, but works by expliciting existing one.
- Its properties are largely independent from the kind of calculus we use (sequent-like, natural deduction-like, Hilbert-like, tableaux-like).
- It can be used to prove strong constructivity of a calculus or theory, although it cannot prove consistency.

Informally, strong constructivity [8] can be defined as follows:

- If we can prove $A \vee B$ then we can prove A or we can prove B, and the information we need to do so, can be effectively found in the proof of $A \vee B$.
- If we can prove $\exists\, x.\, A(x)$ then we can prove $A(t)$ for some term t (if x is a first-order variable) or some formula t (if x is a second-order variable), and the information to derive $A(t)$ can be effectively found in the proof of $\exists\, x.\, A(x)$.

A natural extension of the collection method would be to higher-order logics. In this paper we will give this enhancement, showing that all features of the method are preserved. This is not trivial since we have to deal with impredicative notions, and with incomplete logics.

While we restrict ourselves to the analysis of second-order intuitionistic logic, our results are immediately extendible to any finite-order intuitionistic logic, both predicative and propositional. And most results holding for first-order theories, such as arithmetic, and sets of Harrop formulas, can be immediately lifted to those higher-order logics.

The main result we present here is an optimal proof of strong constructivity for the second-order intuitionistic predicate calculus in the Gentzen-Prawitz style (natural deduction).

This result (strong constructivity) could be drawn as a consequence of the Normalization Theorem [22, 3], but it is not an immediate corollary, and it couldn't be easily generalized to other logics or theories.

In contrast, our method is easy to generalize and it is more economical too, being able to show constructivity of our logic performing at most ω computation steps.

The outline of the paper is as follows:

- In the first section we introduce the second-order intuitionistic calculus, specifying its syntax, inference rules and relevant definitions.
- The second section contains essentials of the collection method.
- In the third section we show that our construction extracts enough information from a proof to ensure that the calculus is constructive. As a corollary, the logic itself is constructive.
- In the final section we discuss consequences of our results.

2 Natural Calculus for Second-Order Logic

In this section our goal is to describe the shape of a calculus for second-order intuitionistic logic. In the usual way [19] we introduce the definitions of terms, formulas and proofs:

-
$$\mathrm{Term} ::= \mathrm{Var_I} \mid \mathrm{Fun}_n(\mathrm{Term}_1, \ldots, \mathrm{Term}_n) \ , \tag{1}$$

  where $\mathrm{Var_I}$ is a denumerable set of first-order variables, and Fun_n is a set of functional symbols with arity n; we adopt the usual convention that Fun_0 is the set of constant symbols.

-
$$\begin{aligned} \mathrm{Formula} ::= {} & \mathrm{Atomic} \mid \mathrm{Var_{II}}(\mathrm{Term}, \ldots, \mathrm{Term}) \mid \\ & \mathrm{Formula} \wedge \mathrm{Formula} \mid \mathrm{Formula} \vee \mathrm{Formula} \mid \\ & \mathrm{Formula} \to \mathrm{Formula} \mid \bot \mid \\ & \forall\, \mathrm{Var_I}.\, \mathrm{Formula} \mid \exists\, \mathrm{Var_I}.\, \mathrm{Formula} \mid \\ & \forall_2\, \mathrm{Var_{II}}.\, \mathrm{Formula} \mid \exists_2\, \mathrm{Var_{II}}.\, \mathrm{Formula} \mid \end{aligned} \tag{2}$$

$$\mathrm{Atomic} ::= \mathrm{Pred}_n(\mathrm{Term}_1, \ldots, \mathrm{Term}_n) \ , \tag{3}$$

  where $\mathrm{Var_{II}}$ is a denumerable set of second-order variables, disjoint from $\mathrm{Var_I}$, and Pred_n is a set of relational symbols with arity n. We assume the usual conventions for bindings, associations and the standard definitions of free, bound variables and substitution [14, 19].

Our calculus is an instance of the standard formulation of the Gentzen-Prawitz system adapted to our syntax. Its set of inference rules is shown in Fig. 1. Let $\mathbf{IL}_2$ be the name of this particular presentation of our logic. It is

Fig. 1. Inference rules for second-order intuitionistic logic

$$\frac{A \quad B}{A \wedge B}\wedge\mathrm{I} \qquad \frac{A \wedge B}{A}\wedge\mathrm{E}_l \qquad \frac{A \wedge B}{B}\wedge\mathrm{E}_r$$

$$\frac{A}{A \vee B}\vee\mathrm{I}_l \qquad \frac{B}{A \vee B}\vee\mathrm{I}_r \qquad \frac{A \vee B \quad \begin{matrix}\cancel{A}\\ \Pi\\ C\end{matrix} \quad \begin{matrix}\cancel{B}\\ \Pi\\ C\end{matrix}}{C}\vee\mathrm{E}$$

$$\frac{\begin{matrix}\cancel{A}\\ \Pi\\ B\end{matrix}}{A \to B}\to\mathrm{I} \qquad \frac{A \quad A \to B}{B}\to\mathrm{E}$$

$$\frac{\bot}{a}\bot\mathrm{E}$$

$$\frac{A(p)}{\forall\, x.\, A(x)}\forall\mathrm{I}\ (*) \qquad \frac{\forall\, x.\, A(x)}{A(t)}\forall\mathrm{E}$$

$$\frac{A(t)}{\exists\, x.\, A(x)}\exists\mathrm{I} \qquad \frac{\exists\, x.\, A(x) \quad \begin{matrix}\cancel{A(p)}\\ \Pi\\ B\end{matrix}}{B}\exists\mathrm{E}^{(**)}$$

$$\frac{A(\rho)}{\forall_2\, \alpha.\, A(\alpha)}\forall_2\mathrm{I}\ (*) \qquad \frac{\forall_2\, \alpha.\, A(\alpha)}{A(B)}\forall_2\mathrm{E}$$

$$\frac{A(B)}{\exists_2\, \alpha.\, A(\alpha)}\exists_2\mathrm{I} \qquad \frac{\exists_2\, \alpha.\, A(\alpha) \quad \begin{matrix}\cancel{A(p)}\\ \Pi\\ B\end{matrix}}{B}\exists_2\mathrm{E}^{(**)}$$

where A,B,C are formulas, a is an atomic formula, t is a term, p and ρ are variables (called *parameters* or *eigenvariables*). In $(*)$ we require that p (ρ) is not free in any open assumption in the deduction of $A(p)$ ($A(\rho)$ respectively), and in $(**)$ we require that p (ρ) is not free in B and in any open assumption in the derivation of B, except $A(p)$, $A(\rho)$. We denote a closed assumption as $\cancel{A}$.

immediate to prove the equivalence of $\mathbf{IL}_2$ to analogous systems which can be found in literature.

We define in the usual way what means derivation:

Definition 1 Derivation. We write $\Gamma \vdash A$ if there is a tree which root is A, which leaves are either closed assumptions or elements of Γ, and which edges are instances of the inference rules of $\mathbf{IL}_2$.

For the sake of brevity we write $\vdash A$ if the set of assumptions is empty.

It is easy to give a semantical proof of consistency for this calculus [22], and it is easy to prove its incompleteness on the standard (i.e. Tarski's) semantics [10]. A Normalization Theorem holds for $\mathbf{IL}_2$, and its proof follows the guidelines for other intuitionistic second-order calculi [19].

A natural way to analyze a proof is to consider its subproofs. A *subproof* of a proof $\prod$ is defined as:

- $\prod$ is a subproof of $\prod$.
- $\prod'$ is a subproof of $\prod$ if it is a subproof of one of its premises.

Because of the nature of the collection method, we will work most of the time with finite sets of proofs, so we prefer to give definitions on sets of proofs.

Notation 2 ($\prec$ Relation) *We write* $\overset{\Gamma}{\underset{A}{\prod}} \prec \mathcal{I}$ *to indicate that* $\overset{\Gamma}{\underset{A}{\prod}}$ *is a subproof of a proof in the set* $\mathcal{I}$.

The collection method takes a finite set of proofs as input, and returns a (possibly infinite) set of formulas as output. We are interested in the output set, which contains the extracted information.

The main property we interested in is to state that the output behaves in a natural way with respect to connectives and quantifiers. Formally speaking:

Definition 3 Pseudo-Truth Set. A set $\mathcal{F}$ of formulas is a pseudo-truth set if and only if

- $A \in \mathcal{F}$ implies $\vdash A$.
- $A \wedge B \in \mathcal{F}$ implies $A \in \mathcal{F}$ and $B \in \mathcal{F}$.
- $A \vee B \in \mathcal{F}$ implies $A \in \mathcal{F}$ or $B \in \mathcal{F}$.
- $A \rightarrow B \in \mathcal{F}$ implies $A \notin \mathcal{F}$ or $B \in \mathcal{F}$.
- $\exists\, x.\, A(x) \in \mathcal{F}$ implies $A(t) \in \mathcal{F}$ for some term t.
- $\exists_2\, \alpha.\, A(\alpha) \in \mathcal{F}$ implies $A(B) \in \mathcal{F}$ for some formula B.

It is worth noting that this definition leads to some pathological examples: the set $\{\exists_2\, \alpha.\, \alpha\}$ is a pseudo-truth set. The witness for $\exists_2\, \alpha.\, \alpha$ is itself! This impredicative behaviour is typical of higher-order logics and it depends on the fact that the substitution operator does not preserve the complexity of formulas, in whatever way we define a complexity measure.

3 The Collection Method

The collection method is a family of procedures with a common structure: informally, they use two operators, Collect and Exp, and compute Collect $\circ\, \mu$Exp, where μ is the least fixed point of its argument.

Since we need to label formulas we collect with their proof, we introduce two operators which project the left or the right component of elements in a set of pairs.

Notation 4 (Left **and** Right **Operators)** *Let $\mathcal{P}$ be a set of pairs:*

$$\text{Left}(\mathcal{P}) = \{x \mid \exists\, y.\, \langle x, y\rangle \in \mathcal{P}\} \ , \tag{4}$$

$$\text{Right}(\mathcal{P}) = \{y \mid \exists\, x.\, \langle x, y\rangle \in \mathcal{P}\} \ . \tag{5}$$

Given a finite set of proofs we can explicit its evident information by collecting conclusions of subproofs. We are interested in provably true formulas. A natural way to perform this process is to define an operator, Coll, as follows:

Definition 5 Coll **and** Collect **Operators.** Let $\mathcal{I}$ be a (possibly infinite) set of proofs, Coll $(\mathcal{I})$ is the smallest set such that, if $\prod_{B}^{\Delta} \prec \mathcal{I}$ and $\Delta \subseteq \text{Left}(\text{Coll}(\mathcal{I}))$, then $\langle B, \prod_{B}^{\Delta}\rangle \in \text{Coll}(\mathcal{I})$;

$$\text{Collect}(\mathcal{I}) = \text{Left}(\text{Coll}(\mathcal{I})) \ . \tag{6}$$

We note that, when $\mathcal{I}$ is finite, Coll $(\mathcal{I})$ and Collect $(\mathcal{I})$ are finite.

A property of the Collect operator is given in the following proposition:

Proposition 6. *If $\mathcal{I}$ is a finite set of proofs and $\phi \in$ Collect $(\mathcal{I})$, then $\vdash \phi$.*

Proof: Immediate, by definition of Coll. We note that the proof of ϕ is obtained "combining" proofs in Right (Coll $(\mathcal{I})$). ■

In general, we are not able to collect a formula which is the conclusion of a parametric subproof. The question now is, how can we extract information from parametric proofs, i.e. deductions containing eigenvariables? The answer is: let's instantiate them! A good instantiation strategy must preserve the closure property implicit in the definition of Coll, the one which makes us able to prove Proposition 6 by composing proofs.

Definition 7 $\forall$I$-$Subst **and** $\forall_2$I$-$Subst **Operators.** Let $\mathcal{I}$ be a finite set of proofs, $\forall$I$-$Subst $(\mathcal{I})$ is the smallest set such that, if

$$\cfrac{\prod_{A(p)}^{\Gamma}(p)}{\forall\, x.\, A(x)}\,\forall\text{I} \prec \mathcal{I}$$

where p is the proper parameter, $\{\forall\, x.\, A(x)\} \cup \Gamma \subseteq \text{Collect}(\mathcal{I})$ and there is a term t such that $A(t) \in \text{Collect}(\mathcal{I})$, then

$$\prod_{A(p)}^{\Gamma}(p{:=}t) \in \forall\text{I}{-}\text{Subst}(\mathcal{I}) \ .$$

The $\forall_2$I$-$Subst operator is defined in the same way, using the $\forall_2$I inference rule.

Definition 8 $\exists$E$-$Subst **and** $\exists_2$E$-$Subst **Operators.** Let $\mathcal{I}$ be a finite set of proofs, $\exists\text{E}-\text{Subst}(\mathcal{I})$ is the smallest set such that, if

$$\dfrac{\overset{\Gamma}{\underset{\exists\, x.\, A(x)}{\prod_1}} \qquad \overset{\Gamma, A(p)}{\underset{B}{\prod_{2(p)}}}}{B}\,\exists\text{E} \prec \mathcal{I}$$

where p is the proper parameter, $\{B\} \cup \Gamma \subseteq \text{Collect}(\mathcal{I})$ and there is a term t such that $A(t) \in \text{Collect}(\mathcal{I})$, then

$$\overset{\Gamma, A(p)}{\underset{B}{\prod_{2(p:=t)}}} \in \exists\text{E}-\text{Subst}(\mathcal{I}) \quad .$$

The $\exists_2$E$-$Subst is defined in a similar way, using the $\exists_2$E inference rule.

Being $\mathcal{I}$ finite, all these set are finite and computable.

Definition 9 Exp **Operator.** Let $\mathcal{I}$ be a finite set of proofs;

$$\begin{aligned} \text{Exp}(\mathcal{I}) = \mathcal{I} &\cup \forall\text{I}-\text{Subst}(\mathcal{I}) \cup \exists\text{E}-\text{Subst}(\mathcal{I}) \cup \\ &\forall_2\text{I}-\text{Subst}(\mathcal{I}) \cup \exists_2\text{E}-\text{Subst}(\mathcal{I}) \quad . \end{aligned} \tag{7}$$

Now we are able to compute the least fixed point of Exp containing $\mathcal{I}$ and to collect the set of true formulas applying Collect to the result. A program for doing this operation is shown in Fig. 2. We note that:

- The least fixed point exists because, calling Ξ the set of all proofs and letting $\mathcal{P}(\Xi)$ be the power-set of Ξ, $\langle \mathcal{P}(\Xi), \subseteq \rangle$ is an algebraic lattice [2, 20].
- Even if the output of our program is a denumerable set, its calculation takes no more than ω steps.

To reason about the properties of the fixed point, we prepare:

Definition 10 Exp***,** Coll* **and** Collect* **Operators.** Let $\mathcal{I}$ be a finite set of proofs,

$$\begin{aligned} \text{Exp}^0(\mathcal{I}) &= \mathcal{I} \\ \text{Exp}^{i+1}(\mathcal{I}) &= \text{Exp}\left(\text{Exp}^i(\mathcal{I})\right) \end{aligned} \tag{8}$$

In the obvious way,

$$\text{Exp}^*(\mathcal{I}) = \bigcup_{i \in \omega} \text{Exp}^i(\mathcal{I}) \tag{9}$$

$$\text{Coll}^*(\mathcal{I}) = \bigcup_{i \in \omega} \text{Coll}\left(\text{Exp}^i(\mathcal{I})\right) \tag{10}$$

$$\text{Collect}^*(\mathcal{I}) = \text{Collect}\left(\text{Exp}^*(\mathcal{I})\right) \tag{11}$$

Lemma 11. *The least fixed point containing $\mathcal{I}$ of the transformation* Exp *is* $\text{Exp}^*(\mathcal{I})$.

Fig. 2. An algorithm to extract information from a set of closed proofs

```
J := ∅
F := ∅
input I
while (I ≠ J)
        begin
          J := I
          F := Collect(I)
          I := Exp(I)
        end
output F
```

Proof: A simple application of the Knaster-Tarski fixed point theorem[20]. ∎

We observe that, by continuity of Exp, since Collect is a monotone map, we get:

Lemma 12. $\text{Collect}^* (\mathcal{I}) = \bigcup_{i \in \omega} \text{Collect}\left(\text{Exp}^i (\mathcal{I})\right)$.

This fact makes trivial to prove correctness of the algorithm in Fig. 2.

As concluding remarks we note:

- $\text{Exp}^* (\mathcal{I})$, $\text{Coll}^* (\mathcal{I})$ and $\text{Collect}^* (\mathcal{I})$ are recursively enumerable, but need not to be finite or recursive.
- $\text{Coll}^* (\mathcal{I})$ is not needed in the computation of $\text{Collect}^* (\mathcal{I})$; it is just a useful device to prove properties about $\text{Collect}^* (\mathcal{I})$.
- We can easily extend the definition of Exp to any finite order calculus which inference rules are close to the ones of $\mathbf{IL}_2$, obtaining the same results. We can confine ourselves to propositional calculi by deleting the ∀I and ∃E inference rules, without substantially affecting our definitions.
- We can safely add new inference rules, provided they are not parametric, and we can add new parametric rules, upon redefining in a proper way the Exp operator.

4 Strong Constructivity

As informally stated in the introduction, a logic is strongly constructive if and only if witnesses for disjunctive and existential formulas could be extracted from their proofs. This concept could be stated in a formal way, but the definitional machinery is quite long and involved, [4]. For our needs, the informal definition is enough, since it is obvious that a strongly constructive logic is constructive in the usual sense. The precise definition is required to show the vice versa (i.e. constructivity implies strong constructivity) does not hold.

In this section we will prove that second-order intuitionistic logic, the one modeled by our calculus, is strongly constructive. This is a simple corollary of the main property of the collection method: Collect* $(\mathcal{I})$ is a pseudo-truth set for any finite set $\mathcal{I}$ of closed proofs. The proof is quite long, so a brief outline is shown in Fig. 3.

Fig. 3. Outline of the proof steps

- We need to assign a complexity measure to every element of Coll* $(\mathcal{I})$.
- Such a measure let us define the notion of *well-given formula*: it is a way to say that an element $\langle A, \prod \rangle$ in Coll* $(\mathcal{I})$ has a relation with the rest of the set which justifies the "truth" of A. For example, $\exists\, x. B(x)$ is well-given if there is a term t such that $B(t)$ is well-given.
- We prove some lemmas about closure properties of Coll* $(\mathcal{I})$. These ones leads us to prove that, if $A \in$ Collect* $(\mathcal{I})$, then A is well-given.
- It follows that Collect* $(\mathcal{I})$ is a pseudo-truth set and that $\mathbf{IL}_2$ is strongly constructive, i.e., second-order intuitionistic logic is constructive. Since the collection procedure takes no more than ω steps, our result is optimal.

Now, it is well known that one of the main technical problems in treating second-order logics is to avoid impredicativity [22]. In order to ensure well-foundness of our principal definition, that of well-given formula, we need some complexity measures.

Definition 13 Propositional Complexity of a Formula. Let ϕ be a formula: its propositional complexity is defined to be the depth of its tree representation with $\bot$ and predicates (i.e. $P(t_1, \ldots, t_n)$, $P \in \mathrm{Pred}_n \cup \mathrm{Var}_{\mathrm{II}}$) as leaves.

Definition 14 Complexity of a Proof. Let $\prod$ be a proof: its complexity is the depth of the associated proof tree.

We note that the complexity of a proof is invariant with respect to substitutions of variables, while the propositional complexity of a formula is invariant only with respect to first-order variables substitutions.

Definition 15 $\forall_2$E-Complexity and $\exists_2$I-Complexity. Let $\prod$ be a proof: its $\forall_2$E-complexity is the negative number such that its absolute value is the maximum number of occurrences of applications of the $\forall_2$E inference rule in a branch; its $\exists_2$I-complexity is the maximum number of occurrences of applications of the $\exists_2$I inference rule in a branch of $\prod$.

We note that the complexity of $\prod$ is an upper bound for its $\exists_2$I-complexity and for the absolute value of its $\forall_2$E-complexity.

It is interesting that the complexity measures over our proofs we need to ensure well-foundness of our principal definition are based on the symmetric rules of our parametric higher-order rules. This is not an accident, but there is a deep relation with impredicativity of higher-order logics.

The complexity of an element of Coll* $(\mathcal{I})$ is defined by a combination of the above:

Definition 16 $\sharp$ Operator. Let $\langle A, \prod\rangle$ be a pair whose first element is a formula, and whose second element is a proof:

$$\sharp(\langle A, \prod\rangle) = \langle a, b, c\rangle$$

where a is the $\forall_2$E-complexity of $\prod$, b is the $\exists_2$I-complexity of $\prod$, and c is the propositional complexity of A.

Definition 17 $\lessdot$ Relation. The relation $\lessdot \subseteq (-\mathcal{N} \times \mathcal{N} \times \mathcal{N})^2$, where $\mathcal{N}$ is the set of natural numbers, is defined as $\langle a, b, c\rangle \lessdot \langle a', b', c'\rangle$ if and only if

- $a < a'$
- $a = a'$ and $b < b'$
- $a = a'$ and $b = b'$ and $c < c'$.

We note that, being equivalent to a lexicographic order, $\lessdot$ is indeed an order relation, and it is total and discrete, i.e. for all x,y such that $x \lessdot y$, the set $\{x \mid x \lessdot z \wedge z \lessdot y\}$ is finite.

As already mentioned, the most important definition we use to prove our result is that of well-given formula. Informally, fixed a set $\mathcal{I}$ of proofs, a formula ϕ is well-given in Coll* $(\mathcal{I})$ if it is collected in Collect* $(\mathcal{I})$ and instances of its immediate subformulas which are in Collect* $(\mathcal{I})$, are well-given, too. If we formalize directly this concept, we would get a non well-founded definition. This is the reason why we introduced complexities. We note that the informal description of well-given formulas has a close relation to the definition of pseudo-truth sets: intuitively, a well-given formula cannot cause problems when proving that Collect* $(\mathcal{I})$ is a pseudo-truth set.

But now it is time to give the formal definition:

Definition 18 Simpler. We say that A is simpler than $\langle B, \prod_1\rangle$, where $A \in$ Collect* $(\mathcal{I})$ and $\langle B, \prod_1\rangle \in$ Coll* $(\mathcal{I})$, if and only if

$$\exists \textstyle\prod_2 . \langle A, \prod_2\rangle \in \text{Coll}^* (\mathcal{I}) \wedge \sharp(\langle A, \prod_2\rangle) \lessdot \sharp(\langle B, \prod_1\rangle) \enspace .$$

Definition 19 Well-Given Formula. Let A be a formula and $\mathcal{I}$ a finite set of proofs, we say that A is well-given (in Coll* $(\mathcal{I})$) if and only if there is a proof $\prod$ such that $\langle A, \prod\rangle \in$ Coll* $(\mathcal{I})$ and

- $A \equiv \bot$ or A is atomic.
- $A \equiv B \wedge C$ and both B and C are well-given and simpler than $\langle A, \prod \rangle$.
- $A \equiv B \vee C$ and either B is well-given and simpler than $\langle A, \prod \rangle$, or C is well-given and simpler than $\langle A, \prod \rangle$.
- $A \equiv B \to C$ and, if B is well-given and simpler than $\langle A, \prod \rangle$, then C is well-given and simpler than $\langle A, \prod \rangle$.
- $A \equiv \forall x. B(x)$ and, for all terms t such that $B(t) \in \text{Collect}^* (\mathcal{I})$, $B(t)$ is well-given and simpler than $\langle A, \prod \rangle$.
- $A \equiv \exists x. B(x)$ and there is a term t such that $B(t)$ is well-given and simpler than $\langle A, \prod \rangle$.
- $A \equiv \forall_2 \alpha. B(\alpha)$ and, for every formula C such that $B(C) \in \text{Collect}^* (\mathcal{I})$ and $B(C)$ is simpler than $\langle A, \prod \rangle$, $B(C)$ is well-given.
- $A \equiv \exists_2 \alpha. B(\alpha)$ and there is a formula C such that $B(C)$ is well-given and simpler than $\langle A, \prod \rangle$.

Proposition 20. *The definition of well-giveness is well-founded.*

Proof: Let us call witnesses for A the set of formulas we need to check for well-giveness in order to ensure that A is well-given. For example, the witnesses for $B \wedge C$ are $\{B, C\}$.

The definition of well-giveness is well-founded if the depth of a recursive expansion of the set of witnesses is finite.

It follows that, defining

$$\mathcal{D} = \{\sharp(\langle B, \textstyle\prod 1 \rangle) \mid \langle B, \textstyle\prod 1 \rangle \in \text{Coll}^* (\mathcal{I})\}$$

if $\langle \mathcal{D}, \lessdot \rangle$ contains no infinite chains, then well-giveness is well-founded.

As noted before, the complexity of a proof is invariant with respect to variable substitution, so the maximum proof complexity of $\text{Right}(\text{Coll}^* (\mathcal{I}))$ is the same as $\mathcal{I}$. But $\mathcal{I}$ is finite. Let us call it α: it is clear that $\beta = \langle -\alpha, 0, 0 \rangle$ is a lower bound for the complexity of every element in $\mathcal{D}$. But $\mathcal{D}$ is a suborder of $\langle -\mathcal{N} \times \mathcal{N} \times \mathcal{N}, \lessdot \rangle$, so it follows that it is total and discrete, and it has β as a lower bound.

If $\beta \in \mathcal{D}$ then it is the minimum of $\mathcal{D}$. Let's suppose that $\beta \notin \mathcal{D}$, and fix $d \in \mathcal{D}$. Let $P_d = \{x \mid x \in \mathcal{D} \wedge x \lessdot d\}$. If $P_d = \emptyset$, d is the minimum of $\mathcal{D}$; if $P_d \neq \emptyset$ then there is $d' \in P_d$; but $|P_d| > |P_{d'}|$ because $|\{x \mid \beta \lessdot x \wedge x \lessdot d\}|$ is finite by discreteness of the ordering $\lessdot$, so, iterating the procedure a finite number of times, we find a minimum in $\mathcal{D}$.

But $\mathcal{D}$ is discrete, and naming γ its minimum, for all x in $\mathcal{D}$, the set $\{y \mid \gamma \lessdot y \wedge y \lessdot x\}$ is finite. In other words, no infinite chain is present in $\mathcal{D}$. It follows that the definition of well-given formula is well-founded. ■

Now we can begin to prove the series of results which state that $\text{Collect}^*(\mathcal{I})$ is a pseudo-truth set: essentially they are closure properties of $\text{Collect}^*(\mathcal{I})$.

Lemma 21. *If* $\prod_A^\Gamma \prec \text{Exp}^*(\mathcal{I})$ *and* $\Gamma \subseteq \text{Collect}^*(\mathcal{I})$, *then* $\langle A, \prod_A^\Gamma \rangle \in \text{Coll}^*(\mathcal{I})$, *and* $A \in \text{Collect}^*(\mathcal{I})$.

Proof: From $\prod_A^\Gamma \prec \text{Exp}^*(\mathcal{I})$ it follows that there is an index j such that $\prod_A^\Gamma \prec \text{Exp}^j(\mathcal{I})$ and, for all $i \geq j$, $\prod_A^\Gamma \prec \text{Exp}^i(\mathcal{I})$.

From $\{B_1, \ldots, B_n\} = \Gamma \subseteq \text{Collect}^*(\mathcal{I})$ it follows that there are indexes $i_1,\ldots,i_n$ such that, for $1 \leq k \leq n$, $B_k \in \text{Collect}\left(\text{Exp}^{i_k}(\mathcal{I})\right)$ and, for all $i \geq i_k$, $B_k \in \text{Collect}\left(\text{Exp}^i(\mathcal{I})\right)$. Let m be the maximum in $j,i_1,\ldots,i_n$, then $\prod_A^\Gamma \prec \text{Exp}^m(\mathcal{I})$ and $\Gamma \subseteq \text{Collect}(\text{Exp}^m(\mathcal{I}))$, so, by definition of the Coll operator, $\langle A, \prod_A^\Gamma \rangle \in \text{Coll}(\text{Exp}^m(\mathcal{I}))$, that is, $\langle A, \prod_A^\Gamma \rangle \in \text{Coll}^*(\mathcal{I})$, implying also that $A \in \text{Collect}^*(\mathcal{I})$. ■

Lemma 22. *If* $\prod_A^\Gamma \prec \text{Exp}^*(\mathcal{I})$ *and, for every B in Γ, B is well-given, then A is well-given.*

Proof: From the preceding lemma we know that $\langle A, \prod_A^\Gamma \rangle \in \text{Coll}^*(\mathcal{I})$, so to prove that A is well-given, we need to prove that it fulfills requirements on its structure. We proceed by induction on the structure of the proofs:

- Assumption: $\prod_A^\Gamma \equiv A$;
  by hypothesis, being $A \in \Gamma$, it is well-given.
- Contradiction: $\prod_A^\Gamma \equiv \dfrac{\prod_\perp^\Gamma}{A}\,\perp\text{E}$;
  but this means A is an atomic formula, so by Definition 19, it is well-given.
- $\wedge$ Introduction: $\prod_A^\Gamma \equiv \dfrac{\prod_C^\Gamma \quad \prod_D^\Gamma}{C \wedge D}\,\wedge\text{I}$;
  by induction hypothesis, we know that C and D are well-given; they are also simpler (see Definition 18) than $\langle C \wedge D, \prod_{C\wedge D}^\Gamma \rangle$ because $\sharp(\langle C \wedge D, \prod_{C\wedge D}^\Gamma \rangle) = \langle a, b, c \rangle$ and, e.g., $\sharp(\langle C, \prod_C^\Gamma \rangle) = \langle a, b, c' \rangle$ with $c' < c$, so by definition, $A \equiv C \wedge D$ is well-given.

- $\wedge$ Elimination: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C \wedge D}^{\Gamma}}{C} \wedge\mathrm{E}_l$ or $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C \wedge D}^{\Gamma}}{D} \wedge\mathrm{E}_r$;
  by induction hypothesis, $C \wedge D$ is well-given, so, by Definition 19, C and D are well-given.

- $\vee$ Introduction: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C}^{\Gamma}}{C \vee D} \vee\mathrm{I}_l$ or $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{D}^{\Gamma}}{C \vee D} \vee\mathrm{I}_r$;
  by induction hypothesis, C or D is well-given; by Definition 17, it is simpler than $\langle A, \prod\limits_{A}^{\Gamma} \rangle$, so it follows that A is well-given.

- $\vee$ Elimination: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C \vee D}^{\Gamma} \quad \prod\limits_{A}^{\Gamma, \cancel{C}}{}_1 \quad \prod\limits_{A}^{\Gamma, \cancel{D}}{}_2}{A} \vee\mathrm{E}$;
  by induction hypothesis, $C \vee D$ is well-given, so C or D is well-given, too.
  If C is well-given, from the induction hypothesis on $\prod_1$, it follows that A is well-given; if D is well-given, we derive that A is well-given from the induction hypothesis on $\prod_2$.
  So, in every case, A is well-given.

- $\rightarrow$ Introduction: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{D}^{\Gamma, \cancel{C}}}{C \rightarrow D} \rightarrow\mathrm{I}$;
  by Definition 19, if C is not well-given then $C \rightarrow D$ is well-given; if C is well-given, then, by induction hypothesis on $\prod$, D is well-given. But D is simpler than $\langle C \rightarrow D, \prod\limits_{A}^{\Gamma} \rangle$ so, by definition, $C \rightarrow D$ is well-given.

- $\rightarrow$ Elimination: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C}^{\Gamma} \quad \prod\limits_{C \rightarrow A}^{\Gamma}}{A} \rightarrow\mathrm{E}$;
  by induction hypothesis, C and $C \rightarrow A$ are well-given, so, by Definition 19, A is well-given.

- $\forall$ Introduction: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C(p)}^{\Gamma}(p)}{\forall\, x.\, C(x)} \forall\mathrm{I}$;
  let t be a term such that, for an appropriate k, $C(t) \in \mathrm{Collect}\left(\mathrm{Exp}^k(\mathcal{I})\right)$, $\Gamma \subseteq \mathrm{Collect}\left(\mathrm{Exp}^k(\mathcal{I})\right)$ and $\prod\limits_{A}^{\Gamma} \prec \mathrm{Exp}^k(\mathcal{I})$, so, by Definition 7, $\prod\limits_{C(p)}^{\Gamma}(p{:=}t) \prec \mathrm{Exp}^{k+1}(\mathcal{I})$, i.e. , $\prod\limits_{C(p)}^{\Gamma}(p{:=}t) \prec \mathrm{Exp}^{*}(\mathcal{I})$.

  As we noted in Definition 14, $(\sharp(\langle C(t), \prod\limits_{C(p)}^{\Gamma}(p{:=}t) \rangle)) = (\sharp(\langle C(p), \prod\limits_{C(p)}^{\Gamma}(p) \rangle)) \lessdot$

$(\sharp(\langle A, \prod_{A}^{\Gamma}\rangle))$, so we deduce that $C(t)$ is simpler than $\langle A, \prod_{A}^{\Gamma}\rangle$. Also we can apply the induction hypothesis to $\prod_{C(p)}^{\Gamma}(p{:=}t)$, and we get that $C(t)$ is well-given. So, by definition, $\forall\, x.\, C(x)$ is well-given.

– $\forall$ Elimination: $\prod_{A}^{\Gamma} \equiv \dfrac{\prod_{\forall\, x.\, C(x)}^{\Gamma}}{C(t)}\,\forall\mathrm{E}$;
by induction hypothesis, $\forall\, x.\, C(x)$ is well-given, and, by definition, because $C(t) \in \mathrm{Collect}^*(\mathcal{I})$, it is well-given, too.

– $\exists$ Introduction: $\prod_{A}^{\Gamma} \equiv \dfrac{\prod_{C(t)}^{\Gamma}}{\exists\, x.\, C(x)}\,\exists\mathrm{I}$;
by induction hypothesis, $C(t)$ is well-given, and, from Definition 16, it is also simpler than $\langle A, \prod_{A}^{\Gamma}\rangle$, so $\exists\, x.\, C(x)$ is well-given.

– $\exists$ Elimination: $\prod_{A}^{\Gamma} \equiv \dfrac{\prod_{\exists\, x.\, C(x)}^{\Gamma} \qquad \prod_{A}^{\Gamma, \cancel{C(p)}}(p)}{A}\,\exists\mathrm{E}$;
by induction hypothesis, $\exists\, x.\, C(x)$ is well-given, so there is a term t such that $C(t)$ is well-given; an immediate consequence of this fact is that $C(t) \in \mathrm{Collect}^*(\mathcal{I})$.

But there must be an index k such that $\{C(t)\} \cup \Gamma \subseteq \mathrm{Collect}\left(\mathrm{Exp}^k(\mathcal{I})\right)$ and $\prod_{A}^{\Gamma} \prec \mathrm{Exp}^k(\mathcal{I})$, so, by Definition 8, $\prod_{A}^{\Gamma, C(p)}(p{:=}t) \prec \mathrm{Exp}^{k+1}(\mathcal{I})$, that is, $\prod_{A}^{\Gamma, C(p)}(p{:=}t) \prec \mathrm{Exp}^*(\mathcal{I})$.

Applying the induction hypothesis to that proof, we get that A is well-given.

– $\forall_2$ Introduction: $\prod_{A}^{\Gamma} \equiv \dfrac{\prod_{C(p)}^{\Gamma}(p)}{\forall_2\, \alpha.\, C(\alpha)}\,\forall_2\mathrm{I}$;
let D be a formula such that $C(D) \in \mathrm{Collect}^*(\mathcal{I})$ and $C(D)$ is simpler than $\langle A, \prod_{A}^{\Gamma}\rangle$, so there is an index k such that $\{C(D)\} \cup \Gamma \subseteq \mathrm{Collect}\left(\mathrm{Exp}^k(\mathcal{I})\right)$, and $\prod_{A}^{\Gamma} \prec \mathrm{Exp}^k(\mathcal{I})$; by Definition 7 it follows that $\prod_{C(p)}^{\Gamma}(p{:=}D) \prec \mathrm{Exp}^{k+1}(\mathcal{I})$, that is, $\prod_{C(p)}^{\Gamma}(p{:=}D) \prec \mathrm{Exp}^*(\mathcal{I})$.

So, by induction hypothesis on that proof, $C(D)$ is well-given. Then, by definition, $\forall_2\, \alpha.\, C(\alpha)$ is well-given.

– $\forall_2$ Elimination: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{\forall_2\,\alpha.\,C(\alpha)}^{\Gamma}}{C(D)}\,\forall_2\mathrm{E}$;

by induction hypothesis, $\forall_2\,\alpha.\,C(\alpha)$ is well-given. But $C(D) \in \mathrm{Collect}^*(\mathcal{I})$ and

$$(\sharp(\langle C(D), \prod_{A}^{\Gamma}\rangle)) \lessdot (\sharp(\langle \forall_2\,\alpha.\,C(\alpha), \prod_{\forall_2\,\alpha.\,C(\alpha)}^{\Gamma}\rangle)) \ ,$$

because the $\forall_2$E-complexity of $\prod\limits_{A}^{\Gamma}$ is lower than the one of $\prod\limits_{\forall_2\,\alpha.\,C(\alpha)}^{\Gamma}$, so $C(D)$ is simpler than $\langle \forall_2\,\alpha.\,C(\alpha), \prod\limits_{\forall_2\,\alpha.\,C(\alpha)}^{\Gamma}\rangle$, and, by Definition 19, $C(D)$ is well-given.

– $\exists_2$ Introduction: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{C(D)}^{\Gamma}}{\exists_2\,\alpha.\,C(\alpha)}\,\exists_2\mathrm{I}$;

by induction hypothesis, $C(D)$ is well-given. But

$$(\sharp(\langle C(D), \prod_{C(D)}^{\Gamma}\rangle)) \lessdot (\sharp(\langle \exists_2\,\alpha.\,C(\alpha), \prod_{A}^{\Gamma}\rangle))$$

because the $\exists_2$I-complexity of $\prod\limits_{C(D)}^{\Gamma}$ is lower, so $C(D)$ is simpler than $\langle A, \prod\limits_{A}^{\Gamma}\rangle$, and, by definition, this implies $\exists_2\,\alpha.\,C(\alpha)$ is well-given.

– $\exists_2$ Elimination: $\prod\limits_{A}^{\Gamma} \equiv \dfrac{\prod\limits_{\exists_2\,\alpha.\,C(\alpha)}^{\Gamma} \quad \prod\limits_{A}^{\Gamma,\not{C(p)}}(p)}{A}\,\exists_2\mathrm{E}$;

by induction hypothesis, we know $\exists_2\,\alpha.\,C(\alpha)$ is well-given, so there is a formula D such that $C(D)$ is well-given, and, in particular $C(D) \in \mathrm{Collect}^*(\mathcal{I})$. So there is an index k such that $\{C(D)\} \cup \Gamma \subseteq \mathrm{Collect}\left(\mathrm{Exp}^k(\mathcal{I})\right)$ and $\prod\limits_{A}^{\Gamma} \prec \mathrm{Exp}^k(\mathcal{I})$; by Definition 8, it follows that $\prod\limits_{A}^{\Gamma,C(p)}(p{:=}D) \prec \mathrm{Exp}^{k+1}(\mathcal{I})$, that is $\prod\limits_{A}^{\Gamma,C(p)}(p{:=}D) \prec \mathrm{Exp}^*(\mathcal{I})$.

By induction hypothesis on this proof, A is well-given.

■

Lemma 23. $A \in \mathrm{Collect}^*(\mathcal{I})$ *implies* A *is well-given.*

Proof: Let's define a relation between formulas in $\mathrm{Collect}^*(\mathcal{I})$ and $\mathcal{N}$: we write $B\ R_j\ m$ if and only if

– $\prod\limits_{B} \prec \mathrm{Exp}^j(\mathcal{I})$ without undischarged assumptions and $m = 0$.

- $\prod_{B}^{C_1,\ldots,C_n} \prec \mathrm{Exp}^j(\mathcal{I})$, $\{C_1,\ldots,C_n\} \subseteq \mathrm{Collect}\left(\mathrm{Exp}^j(\mathcal{I})\right)$, $C_1 \; R_j \; m_1, \ldots,$ $C_n \; R_j \; m_n$ and $m = 1 + \max\{m_i \mid 1 \leq i \leq n\}$.

Let d_j be the function defined as $d_j(B) = \min\{m \mid B \; R_j \; m\}$.

By induction on $d_j(B)$ we prove that, for all $B \in \mathrm{Collect}\left(\mathrm{Exp}^j(\mathcal{I})\right)$, B is well-given:

- $d_j(B) = 0$; then $\prod_B \prec \mathrm{Exp}^j(\mathcal{I})$, so $\prod_B \prec \mathrm{Exp}^*(\mathcal{I})$, and, by Lemma 22, B is well-given.
- $d_j(B) > 0$; then $\prod_{B}^{C_1,\ldots,C_n} \prec \mathrm{Exp}^j(\mathcal{I})$, with $\{C_1,\ldots,C_n\} \subseteq \mathrm{Collect}\left(\mathrm{Exp}^j(\mathcal{I})\right)$ and, for $1 \leq i \leq n$, $d_j(C_i) \leq d_j(B)$, so $\prod_{B}^{C_1,\ldots,C_n} \prec \mathrm{Exp}^*(\mathcal{I})$; by induction hypothesis, $C_1,\ldots,C_n$ are well-given, and, by Lemma 22, B is well-given.

Now, by hypothesis, $A \in \mathrm{Collect}^*(\mathcal{I})$, that is $A \in \mathrm{Collect}\left(\mathrm{Exp}^j(\mathcal{I})\right)$ for some j, and by the above result, A is well-given. ■

Theorem 24. $\mathrm{Collect}^*(\mathcal{I})$ *is a pseudo-truth set.*

Proof: Let $A \in \mathrm{Collect}^*(\mathcal{I})$, by Proposition 6 and noting that there is an index j such that $A \in \mathrm{Collect}\left(\mathrm{Exp}^j(\mathcal{I})\right)$, it follows that $\vdash A$. By Lemma 23, A is well-given so it fulfills the requirements of Definition 3 on the structure of formulas. ■

Corollary 25. *Second-order intuitionistic logic is strongly constructive.*

Proof: Let C be a closed formula such that $\vdash C$:

- Let $C \equiv A \vee B$, then there is a proof $\prod_{A\vee B}$: by Theorem 24, $\mathrm{Collect}^*\left(\{\prod_{A\vee B}\}\right)$ contains $A \vee B$ and also A or B. But, being a pseudo truth set, $\vdash A$ or $\vdash B$, and the corresponding proof is composed combining proofs in $\mathrm{Exp}^*\left(\{\prod_{A\vee B}\}\right)$.
- Let $C \equiv \exists\, x.\, A(x)$ and consider $\prod_{\exists\, x.\, A(x)}$: $\mathrm{Collect}^*\left(\{\prod_{\exists\, x.\, A(x)}\}\right)$ is a pseudo truth set so it contains $\exists\, x.\, A(x)$ and $A(t)$ for some term t; more than this, a proof of $A(t)$ can be found composing proofs in $\mathrm{Exp}^*\left(\{\prod_{\exists\, x.\, A(x)}\}\right)$.
- Let $C \equiv \exists_2\, \alpha.\, A(\alpha)$: in an analogous way as in the $\exists$ case, there is a B such that $A(B) \in \mathrm{Collect}^*\left(\{\prod_{\exists_2\, \alpha.\, A(\alpha)}\}\right)$ and its proof is formed composing proofs in $\mathrm{Exp}^*\left(\{\prod_{\exists_2\, \alpha.\, A(\alpha)}\}\right)$.

■

5 Conclusions

We have extended the collection method to second-order logic, showing that it retains all its relevant properties.

The real difference with the first-order version lies in an explicit, and intricate, definition of complexities in order to control impredicativity in the definition of well-given formulas. In fact, having the right definitions, Lemmata 21, 22 and 23 are straightforward. In some way, the proof path does not change raising the order of the logic.

What is surprising, is Definition 16 because of the symmetry between higher-order expansion operators $\forall_2\text{I}-\text{Subst}$ and $\exists_2\text{E}-\text{Subst}$ and their symmetric complexities over $\exists_2\text{I}$ and $\forall_2\text{E}$ rules. This is a point, we think, that needs further investigations.

Of course this result proves also that first-order logic, first-order propositional logic and second order propositional logic are strongly constructive: just drop cases regarding $\forall$, $\exists$, $\forall_2$, $\exists_2$ in the proper way in the proof of Lemma 22.

Our approach could be easily extended to any finite order intuitionistic logic (and to their propositional version): we have to define a new complexity operator following the heuristic of symmetry we underlined above, and then the proof comes out naturally. It is easy to show [4] that any logic could be extended with an arbitrary set of Harrop formulas as axioms without affecting its strongly constructive character. It is also possible without great effort to prove that any finite-order intuitionistic logic extended with Heyting arithmetic remains strongly constructive. Essentially the part of the proof regarding arithmetics is almost the same as in the first-order case [17].

As a first instance, we plan to extend this result to many higher-order intermediate logics [18, 1, 5, 6, 7]. We are also trying to extend this result to the full higher-order intuitionistic logic, that is, to a logic whose order is not finite. We think in the future it is worth investigating the possible extensions to linear [11, 21], modal [9] and temporal logics [15].

References

1. A. Avellone, C. Fiorentini, P. Mantovani, and P. Miglioli. On maximal intermediate predicate constructive logics. *Studia Logica*, 57:373, 1996.
2. H. P. Barendregt. *The Lambda-Calculus: Its Syntax and Semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics*. North Holland Publ. Co., Amsterdam, 2nd edition, 1984.
3. A. G. Dragalin. *Mathematical Intuitionism*, volume 67 of *Translations of Mathematical Monographs*. American Mathematical Society, 1987.
4. M. Ferrari. *Strongly Constructive Formal Systems*. PhD thesis, Dipartimento di Scienze dell'Informazione – Università degli Studi di Milano, Milan, 1996. `http://eolo.usr.dsi.unimi.it/~ferram/thesis.ps.gz`.
5. M. Ferrari and P. Miglioli. Counting the maximal intermediate constructive logics. *Journal of Symbolic Logic*, 58(4):1365–1401, 1993.

6. M. Ferrari and P. Miglioli. A method to single out maximal propositional logics with the disjunction property I. *Annals of Pure and Applied Logic*, 76:1–46, 1995.
7. M. Ferrari and P. Miglioli. A method to single out maximal propositional logics with the disjunction property II. *Annals of Pure and Applied Logic*, 76:117–168, 1995.
8. M. Ferrari, P. Miglioli, and M. Ornaghi. Strongly constructive formal systems. *Annals of Pure and Applied Logic*, 1997. Submitted.
9. G. Fischer-Servi. On modal logic with an intuitionistic base. *Studia Logica*, 36(3):141–149, 1977.
10. J. Y. Girard. *Interprétation Fonctionelle et Élimination des Coupures de l'Arithmétique d'Ordre Supérieur.* Thèse de doctorat d'état, Université Paris VII, Paris, June 1972.
11. J. Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
12. S. C. Kleene. *Introduction to Metamathemathics.* North Holland Publ. Co., Amsterdam, 1952.
13. G. Kreisel. Mathematical significance of consistency proofs. *Journal of Symbolic Logic*, 23(2):155–182, 1958.
14. D. Leivant. Higher-order logic. In D. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming: Deduction Methodologies*, volume II, pages 229–321. Clarendon Press, Oxford, 1994.
15. Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems.* Springer Verlag, Berlin, 1992.
16. P. Miglioli and M. Ornaghi. A logically justified model of computation I. *Fundamenta Informaticae: Annales Societatis Mathematicae Polonae*, IV(1):151, 1981.
17. P. Miglioli and M. Ornaghi. A logically justified model of computation II. *Fundamenta Informaticae: Annales Societatis Mathematicae Polonae*, IV(2):277, 1981.
18. H. Ono. A study of intermediate predicate logics. Publications of the Research Institute for Mathematical Sciences 8, Kyoto University, 1972.
19. H. Schwichtenberg and A. S. Troelstra. *Basic Proof Theory*, volume 43 of *Cambridge Tracts in Theoretical Computer Science.* Cambridge University Press, 1996.
20. R. Smullyan. *Diagonalization and Self-Reference*, volume 27 of *Oxford Logic Guides.* Clarendon Press, 1994.
21. A. S. Troelstra. *Lectures on Linear Logic.* CSLI Lecture Notes No. 29. Center for the Study of Language and Information, Stanford University, 1992.
22. A. S. Troelstra and D. van Dalen. *Constructivism in Mathematics, An Introduction*, volume 121,123 of *Studies in Logic and the Foundations of Mathematics.* North Holland Publ. Co., Amsterdam, 1988.

This article was processed using the LaTeX macro package with LLNCS style