On Completeness of Program Synthesis Systems

Andrei Voronkov[1]

ECRC, Arabellastr.17, 8000 Munich 81, Germany (voronkov@ecrc.de)*

Abstract. This paper addresses a question of completeness of systems extracting programs from proofs. We consequently consider specifications of total functions, partial functions, predicates and partial predicates. Our aim is to investigate the problem of existence of a logical system which allows to extract all total recursive function (all partial recursive functions, all decidable predicates) from proofs in this logical system. Another natural question is whether such systems exist among generally known systems. We prove that it is impossible to extract programs for all total recursive functions from the proofs of specifications of the form $\forall\bar{x}\exists y\mathrm{Out}(x,y)$ and it is impossible to extract algorithms deciding all decidable predicates from the proofs of the specifications of the form $\forall\bar{x}(P(\bar{x})\vee\neg P(\bar{x}))$. Concerning specifications of the partial functions and the partial predicates we prove that in the system **HA+MP** consisting from the intuitionistic arithmetic with the Markov's principle it is possible to extract all partial recursive functions from the specifications of the form $\forall\bar{x}(\mathrm{In}(\bar{x})\supset\exists y\mathrm{Out}(\bar{x},y))$ and algorithms deciding all decidable predicates from the specifications of the form $\forall\bar{x}(\mathrm{In}(\bar{x})\supset(P(\bar{x})\vee\neg P(\bar{x})))$.

1 Introduction

There are many approaches to extracting programs from constructive proofs. Many papers on programming in logic address the question about the logical power of the proof systems intended for extracting programs from proofs. It means provability (or non-provability) of some principles of constructive mathematics. Another interesting question is to investigate the intensional behavior of the proof systems, for example answering the question on the relationship between the complexity of some algorithm and the complexity of programs which can be extracted from proofs and which compute the same function.

Our paper has a more modest aim - to investigate the extensional behavior of proof systems for program extraction: if f is a total recursive function (a partial recursive function, a decidable predicate), is it possible to extract a program for computing f from some proof. Of course the answer in the case of one particular function or predicate is "yes" - it is possible to extract it from a proof in some suitable calculus. But it is interesting to find logic in which we can extract all possible programs (at least extensionally) from proofs. In other words, our aim is to investigate cases when there is a logical system which allows to extract all total recursive functions (all partial recursive functions, all decidable predicates) from proofs in this

* On leave from the International Laboratory of Intelligent Systems (SINTEL), 630090, Universitetski Prospect 4, Novosibirsk 90, Russia.

logical system. Another interesting question is whether such systems exist among some known systems.

It is not easy to answer these questions in the general case because the notion of an algorithm or of a program is rather vague. So we shall concentrate on programs over the domain for which such a notion exists and had been intensively investigated - the domain of the natural numbers. From now on we restrict our considerations on the natural numbers and assume that all logics that we discuss use the first order arithmetical language $\langle 0, s, +, \times, = \rangle$. Nevertheless the ideas of the proofs from this paper are very general and applicable to other theories in different first order languages or even to type theories. We shall use $\mathbf{N}$ to denote the set of natural numbers.

We shall also freely use the overbar notation: for example, $\bar{e}$ will stand for the vector $e_1, \ldots, e_n$ and if we use $\bar{e}$ then e_i is the ith component of $\bar{e}$.

An Inference rule is any expression of the form $\varphi_1, \ldots, \varphi_n/\varphi$, where $\varphi_1, \ldots, \varphi_n, \varphi$ are formulae. Axioms are inference rules with $n = 0$. In what follows logic will state for any decidable set of inference rules containing all rules of the intuitionistic predicate calculus. (It means that there exists an algorithm which recognizes whether a given inference rule belongs to the logic or not). Provability in logic $\mathfrak{L}$ will be denoted by $\vdash_{\mathfrak{L}}$ or $\mathfrak{L}\vdash$.

A formula φ is closed iff it has no free variables.

Let m be a natural number. Then $\underline{m}$ will state for the numeral $s^n(0)$ representing this number in arithmetic. For any particular logic $\mathfrak{L}$, let $\nu_{\mathfrak{L}}$ be some fixed Godel enumeration of the set of all proofs in $\mathfrak{L}$. If $\mathfrak{L}$ is logic and φ is a formula then $\mathfrak{L} + \varphi$ stays for the logic obtained from $\mathfrak{L}$ by adding φ as an axiom.

The set Har of Harrop formulae[1] is defined as follows.

1. Any atomic formula is a Harrop formula;
2. If φ, ψ are Harrop formulae and χ is an arbitrary formula then the formulae $\varphi \wedge \psi$, $\forall x \varphi$, $\chi \supset \psi$, $\neg \chi$ are Harrop formulae.

Let f be a total function on natural numbers. The formula $\varphi(\bar{x}, y)$ specifies f iff for any tuple $\bar{m}, n$ of natural numbers $f(\bar{m}) = n$ iff $\varphi(\underline{\bar{m}}, \underline{n})$.

Let f be a partial function on natural numbers. A pair of formulae $\psi(\bar{x})$, $\varphi(\bar{x}, y)$ specifies f iff for any tuple $\bar{m}, n$ of natural numbers we have

1. If $f(\underline{\bar{m}})$ is defined then $\psi(\underline{\bar{m}})$;
2. If $\psi(\underline{\bar{m}})$ then $f(\bar{m}) = n$ iff $\varphi(\underline{\bar{m}}, \bar{n})$.

Let P be a relation on natural numbers. A formula $\varphi(\bar{x})$ specifies P iff for any tuple $\bar{m}$ of natural numbers $P(\bar{m})$ iff $\varphi(\underline{\bar{m}})$.

Via $\mathbf{N}$ we denote the set of all natural numbers. Let P be a relation on natural numbers and $S \subseteq \mathbf{N}$. A pair of formulae $\psi(\bar{x})$, $\varphi(\bar{x})$ specifies P on the set S iff for any tuple m of natural numbers

1. if $\bar{m} \in S$ then $\psi(\underline{\bar{m}})$;
2. if $\psi(\underline{\bar{m}})$ then $P(\bar{m})$ iff $\varphi(\underline{\bar{m}})$.

2 Total functions

We say that a logic $\mathfrak{L}$ has the $\mathrm{PAP_t}$-property (the proofs-as-programs property for total functions) iff there exists a total recursive function ε such that if n is Gödel number of a proof Π in $\mathfrak{L}$ of a closed formula $\forall\bar{x}\exists y\mathrm{Spec}(\bar{x},y)$ then $\varepsilon(n)$ is a number of a total recursive function f such that for any tuple $\bar{m}$ of natural numbers $\mathrm{Spec}(\underline{\bar{m}}, f(\underline{\bar{m}}))$.

Such logic is t-complete iff for every total recursive function f there exists a formula $\mathrm{Spec}(\bar{x},y)$ which specifies this function and a proof Π of the formula $\forall x\exists y\mathrm{Spec}(\bar{x},y)$ in $\mathfrak{L}$.

The $\mathrm{PAP_t}$-property property means that one can extract total recursive functions from proofs. If logic $\mathfrak{L}$ has $\mathrm{PAP_t}$-property then ε is an algorithm which extracts programs from proofs (a program extractor). t-completeness means that we can extract *all* total recursive functions using the extracting algorithm ε.

Theorem 1. *There is no t-complete logic.*

Proof. Assume that such logic $\mathfrak{L}$ exists. Then using $\nu_{\mathfrak{L}}$ and the extracting algorithm ε one can construct an effective enumeration of the set of all total recursive functions. But it is generally known that such an enumeration does not exist.

From results of the next section follows that if we change definitions then it becomes possible to extract all total recursive functions in the following sense:

> There exists a logic $\mathfrak{L}$ and an algorithm ε such that for any total recursive function f there exists a Harrop formula φ_f which is true on natural numbers such that
> 1. ε is an extracting algorithm for $\mathfrak{L}+\varphi_f$;
> 2. There exists a formula $\mathrm{Spec}(\bar{x},y)$ specifying f such that $\forall x\exists y\mathrm{Spec}(\bar{x},y)$ is provable in $\mathfrak{L}+\varphi_f$.

3 Partial functions

We say that logic $\mathfrak{L}$ has the $\mathrm{PAP_p}$-property (proofs-as-programs property for partial functions) iff there exists a total recursive function ε such that if n is a number of a proof Π in $\mathfrak{L}$ of a closed formula $\forall\bar{x}(\mathrm{In}(\bar{x})\supset\mathrm{Out}(\bar{x},y))$, where $\mathrm{In}(\bar{x})$ is a Harrop formula, then $\varepsilon(n)$ is equal to a Kleene number of a partial recursive function f such that for any tuple $\bar{m}$ of natural numbers if $\mathrm{In}(\underline{\bar{m}})$ then $f(\bar{m})$ is defined and $\mathrm{Out}(\underline{\bar{m}}, f(\underline{\bar{m}}))$.

Such logic is p-complete iff for every partial recursive function f there exists a pair of formulae $\mathrm{In}(\bar{x})$, $\mathrm{Out}(\bar{x},y)$, where $In(\bar{x})$ is a Harrop formula, which specifies this function and a proof Π of the formula $\forall\bar{x}(\mathrm{In}(\bar{x})\supset\exists y\mathrm{Out}(\bar{x},y))$ in $\mathfrak{L}$.

The Markov's principle for a formula φ is the formula $\forall x(\varphi(x)\vee\neg\varphi(x))\supset(\neg\neg\exists x\varphi(x)\supset\exists x\varphi(x))$. Let **HA** be the intuitionistic (Heyting) arithmetic. **HA+MP** will stay for the calculus obtained from the intuitionistic arithmetic by adding all instances of the Markov's principle as axioms.

We say that logic $\mathfrak{L}$ has the <u>$\exists$-property</u>[2] iff for any closed formula $\exists x\varphi(x)$ from $\vdash_{\mathfrak{L}}\exists x\varphi(x)$ follows that $\vdash_{\mathfrak{L}}\varphi(\underline{n})$ for some natural number n. Logic $\mathfrak{L}$ has the <u>disjunction property</u> iff for closed formulas φ,ψ from $\vdash_{\mathfrak{L}}\varphi\vee\psi$ follows $\vdash_{\mathfrak{L}}\varphi$ or $\vdash_{\mathfrak{L}}\psi$.

Lemma 2. *If χ is a true Harrop formula then* $\mathbf{HA}+\mathbf{MP}+\chi$ *has the disjunction property and the $\exists$-property.*

Proof. Let $\mathfrak{L}=\mathbf{HA}+\mathbf{MP}+\chi$. The proof is based on Kleene's characterization of logics possessing these properties [2].

First we introduce some definitions similar to those of [2]. We introduce a new relation $|$ between sets of formulae and formulae in the following way. During the definition we assume that $T|\!\vdash\varphi$ means $T|\varphi$ and $T\vdash\varphi$, where $\vdash$ stands for the provability in the intuitionistic logic.

1. For atomic formulae φ, $T|\varphi$ iff $T\vdash\varphi$;
2. $T|\varphi\wedge\psi$ iff $T|\varphi$ and $T|\psi$;
3. $T|\varphi\vee\psi$ iff $T|\!\vdash\varphi$ or $T|\!\vdash\psi$;
4. $T|\varphi\supset\psi$ iff from $T|\!\vdash\varphi$ follows $T|\psi$;
5. $T|\neg\varphi$ iff not $T|\!\vdash\varphi$;
6. $T|\forall x\varphi(x)$ iff for every natural number n, $T|\varphi(\underline{n})$;
7. $T|\exists x\varphi(x)$ iff for some natural number n, $T|\!\vdash\varphi(\underline{n})$.

The relation $|\!\vdash$ is identical to Kleene's slash from [2] except for minor changes for the quantifier cases.

As in [2] it is possible to prove that logic obtained from intuitionistic logic by adding a set of formulae S as axioms has the disjunction property and the $\exists$-property iff for any $\varphi\in S$ we have $S|\varphi$. Let $\mathfrak{L}$ be logic obtained from the intuitionistic predicate calculus by adding as axioms all instances of the induction scheme, the Markov's principle, and some number of true Harrop formulae. We prove that $\mathfrak{L}$ satisfies this property.

1. $\mathfrak{L}|\varphi(0)\supset(\forall x(\varphi(x)\supset\varphi(s(x)))\supset\forall x\varphi(x))$. To prove it we assume $\mathfrak{L}|\!\vdash\varphi(0)$ and $\mathfrak{L}|\!\vdash\forall x(\varphi(x)\supset\varphi(s(x)))$. Then trivially for any $n\in\mathbf{N}$ we have $\mathfrak{L}|\!\vdash\varphi(\underline{n})$ and hence $\mathfrak{L}|\forall x\varphi(x)$.
2. $\mathfrak{L}|\forall x(\varphi(x)\vee\neg\varphi(x))\supset(\neg\neg\exists x\varphi(x)\supset\exists x\varphi(x))$. To prove it we assume $\mathfrak{L}|\!\vdash\forall x(\varphi(x)\vee\neg\varphi(x))$ and $\mathfrak{L}|\!\vdash\neg\neg\exists x\varphi(x)$. From $\mathfrak{L}|\!\vdash\neg\neg\exists x\varphi(x)$ follows $\mathfrak{L}\vdash\neg\neg\exists x\varphi(x)$. Since all formulae provable in $\mathfrak{L}$ are true in the standard model of arithmetic, then for some $n\in\mathbf{N}$ we have $\varphi(\underline{n})$. From $\mathfrak{L}|\!\vdash\forall x(\varphi(x)\vee\neg\varphi(x))$ we have that for every $n\in\mathbf{N}$ $\mathfrak{L}|\!\vdash\varphi(\underline{n})\vee\neg\varphi(\underline{n})$. Hence for every $n\in\mathbf{N}$ we have $\mathfrak{L}|\!\vdash\varphi(\underline{n})$ or $\mathfrak{L}|\!\vdash\neg\varphi(\underline{n})$. Let $m\in\mathbf{N}$ be such that $\varphi(\underline{m})$. Then $\mathfrak{L}|\!\vdash\neg\varphi(\underline{m})$ is impossible since only true formulae are provable in $\mathfrak{L}$. So we have $\mathfrak{L}|\!\vdash\varphi(\underline{m})$ and hence $\mathfrak{L}|\exists x\varphi(x)$.
3. $\mathfrak{L}|\varphi$ where φ is a true Harrop formula. As in [2] it is easy to prove that for any Harrop formulae ψ from $\mathfrak{L}\vdash\psi$ follows $\mathfrak{L}|\psi$.

Theorem 3. $\mathbf{HA}+\mathbf{MP}$ *has the* $\mathrm{PAP_p}$*-property.*

[2] In the language of arithmetic it is usually called *the numerical existence property*

Proof. Let Π be a proof in $\mathfrak{L}$ of a closed formula of the form $\forall\bar{x}(\mathrm{In}(\bar{x})\supset\exists y\mathrm{Out}(\bar{x},y))$, where $\mathrm{In}(\bar{x})$ is a Harrop formula. It is sufficient to show how to effectively construct by this proof a partial recursive function f meeting this specification. Let $\bar{m}$ be a tuple of natural numbers. We can effectively enumerate all proofs in the logic $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\underline{\bar{m}})$. If there exists a proof in $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\underline{\bar{m}})$ of a formula of the form $\mathrm{Out}(\underline{\bar{m}},\underline{n})$ then we let $f(\bar{m})=n$, where n is taken from the first such proof, otherwise $f(\bar{m})$ is not defined.

It is easy to see that f is effectively constructed and hence partial recursive. To prove that it meets the specification assume $\mathrm{In}(\underline{\bar{m}})$. Then $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\underline{\bar{m}})\vdash \exists x\mathrm{Out}(\underline{\bar{m}},x)$. Applying Lemma 2 we obtain that $f(\bar{m})$ is defined. By definition of f we have $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\underline{\bar{m}})\vdash\mathrm{Out}(\underline{\bar{m}},\underline{f(\bar{m})})$. Finally note that in $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\underline{\bar{m}})$ only true formulae are provable.

Before proving completeness of $\mathbf{HA{+}MP}$ we introduce some definitions and lemmata.

Let $C_n(x_1,\ldots,x_{n+1})$, $n=2,3\ldots$ be the formulae representing functions for enumeration of the tuples of natural numbers defined in the following way.

$$C_2(x,y,z)\Leftrightarrow 2\times z=(x+y)\times(x+y)+3\times x+y;$$

$$C_{n+1}(\bar{x},y,z)\Leftrightarrow\exists u(C_n(\bar{x},u)\wedge C_2(u,y,z))\text{ for }n\geq 2.$$

Lemma 4. *The following formulas are provable in* $\mathbf{HA}$*:*

1. $\forall\bar{x}yz(C_n(\bar{x},y)\wedge C_n(\bar{x},z)\supset y=z)$;
2. $\forall\bar{x}\bar{y}z(C_n(\bar{x},z)\wedge C_n(\bar{y},z)\supset\bar{x}=\bar{y})$;
3. $\forall\bar{x}\exists yC_n(\bar{x},y)$;
4. $\forall y\exists\bar{x}C_n(\bar{x},y)$.

Proof. Immediate by induction.

Lemma 5. *If φ is a quantifier-free formula of arithmetic and $\bar{x}$ are all variables of φ then the formula $\forall x(\varphi(\bar{x})\vee\neg\varphi(\bar{x}))$ is provable in* $\mathbf{HA}$.

Proof. Trivial.

Theorem 6. $\mathbf{HA{+}MP}$ *is p-complete.*

Proof. Let $f(\bar{x})$ be a partial recursive function. By the Matijasevič's theorem [3] there exists a quantifier-free formula $F(\bar{x},\bar{y},z)$ of the arithmetical language such that for any tuple $\bar{m},n$ of natural numbers $f(\bar{m})=n$ iff $\exists\bar{y}(F(\underline{\bar{m}},\bar{y},\underline{n}))$. Let $G(\bar{x},u)\Leftrightarrow\exists\bar{y}z(F(\bar{x},\bar{y},z)\wedge C_n(\bar{y},z,u))$ and $\mathrm{In}(\bar{x})\Leftrightarrow\neg\neg\exists uG(\bar{x},u)$. Let $\mathrm{Out}(\bar{x},z)\Leftrightarrow\exists\bar{y}F(\bar{x},\bar{y},z)$. We prove that the pair $\mathrm{In}(\bar{x}),\mathrm{Out}(\bar{x},z)$ satisfies the conditions of the definition of p-completeness.

From the following equivalencies we see that the pair $\mathrm{In}(\bar{x}),\mathrm{Out}(\bar{x},z)$ specifies the function f:

$$\begin{aligned}&\mathrm{In}(\bar{x})\Leftrightarrow\\&\exists uG(\bar{x},u)\Leftrightarrow\\&\exists\bar{y}zu(F(\bar{x},\bar{y},z)\wedge C_n(\bar{y},z,u))\Leftrightarrow\\&\exists\bar{y}zF(\bar{x},\bar{y},z)\Leftrightarrow\\&\exists v(f(\bar{x})=v).\end{aligned}$$

It remains to prove that $\mathbf{HA}+\mathbf{MP}\vdash\forall\bar{x}(\mathrm{In}(\bar{x})\supset\exists y\mathrm{Out}(\bar{x},y))$. Using Lemmata 4,5 it is possible to prove that $\mathbf{HA}\vdash\forall\bar{x}u(G(\bar{x},u)\vee\neg G(\bar{x},u))$. Applying the Markov's principle we have $\mathbf{HA}+\mathbf{MP}+\neg\neg\exists uG(\bar{x},u)\vdash\exists uG(\bar{x},u)$ or equivalently $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\bar{x})\vdash\exists uG(\bar{x},u)$. From this immediately follows $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\bar{x})\vdash\exists yG(\bar{x},y)$ or $\mathbf{HA}+\mathbf{MP}\vdash\forall\bar{x}(\mathrm{In}(\bar{x})\supset\exists y\mathrm{Out}(\bar{x},y))$.

The next theorem shows that the use of the Markov's principle is essential.

Theorem 7. **HA** *is not p-complete.*

Proof. Assume that **HA** is p-complete. Let f be any partial recursive function which can not be extended to a total recursive function. Let $\mathrm{In}(\bar{x}),\mathrm{Out}(\bar{x},y)$ specifies f, $\mathrm{In}(\bar{x})$ be a Harrop formula, and $\mathbf{HA}\vdash\forall\bar{x}(\mathrm{In}(\bar{x})\supset\exists y\mathrm{Out}(\bar{x},y))$. We define a general recursive function g in the following way. Let $\bar{m}$ be a tuple of natural numbers. Then $\mathbf{HA}+\mathrm{In}(\underline{\bar{m}})\vdash\exists y\mathrm{Out}(\underline{\bar{m}},y)$. Since $\mathrm{In}(\bar{x})$ is a Harrop formula, $\mathbf{HA}+\mathrm{In}(\underline{\bar{m}})$ has the $\exists$-property. So there is some $n\in\mathbf{N}$ such that $\mathbf{HA}+\mathrm{In}(\underline{\bar{m}})\vdash\mathrm{Out}(\underline{\bar{m}},\underline{n})$. We let $g(\bar{m})=n$ for the first such n (in the enumeration of all proofs in $\mathbf{HA}+\mathrm{In}(\underline{\bar{m}})$). By the construction the function g is a total recursive function. Assume that $f(\bar{m})$ is defined. Then $\mathrm{In}(\underline{\bar{m}})$ and hence $\mathrm{Out}(\underline{\bar{m}},\underline{g(\bar{m})})$. From this follows that $f(\bar{m})=g(\bar{m})$. This contradicts to the assumption that f can not be extended to a total recursive function.

4 Predicates

With minor modifications the results of the two previous sections can be generalized to the predicate case.

We say that logic $\mathfrak{L}$ has the <u>PAP$_{\mathrm{r}}$-property</u> (proofs-as-programs property for relations) iff there exists a total recursive function ε such that if n is a number of a proof Π in $\mathfrak{L}$ of a closed formula of the form $\forall\bar{x}(\varphi(\bar{x})\vee\neg\varphi(\bar{x}))$ then $\varepsilon(n)$ is a number of a general recursive function f such that for any tuple $\bar{n}$ of natural numbers $f(\bar{n})=0$ iff $\varphi(\underline{\bar{n}})$.

Such logic is <u>r-complete</u> iff for every decidable predicate P there exists a formula $\varphi(\bar{x})$ which specifies this predicate and a proof Π of the formula $\forall\bar{x}(\varphi(\bar{x})\vee\neg\varphi(\bar{x}))$ in $\mathfrak{L}$.

As in Section 2 we can prove the following statement

Theorem 8. *There is no r-complete logic.*

Proof. The same as for Theorem 1. We note that if such a logic exists we can effectively construct a computable enumeration of the class of all total recursive functions with values from the set $\{0,1\}$, which is impossible.

r-completeness means that it is possible to extract programs deciding all decidable predicates from proofs of specifications of the form $\forall\bar{x}(P(\bar{x})\vee\neg P(\bar{x}))$ in one logic. Using the notion of the specification on set we can extract all decidable predicates from the specifications of the form $\forall\bar{x}(\mathrm{In}(\bar{x})\supset(P(\bar{x})\vee\neg P(\bar{x})))$.

Theorem 9. *Let* $\mathrm{In}(\bar{x})$ *be a Harrop formula of the arithmetical language and* $\mathbf{HA}+\mathbf{MP}\vdash\forall\bar{x}(\mathrm{In}(\bar{x})\supset(P(\bar{x})\vee\neg P(\bar{x})))$. *Then using a proof of this formula in* $\mathbf{HA}+\mathbf{MP}$ *one can effectively find a partial recursive function* f *such that if* $\mathrm{In}(\bar{x})$ *then* $f(\bar{x})$ *is defined and* $f(\bar{x})=0$ *iff* $P(\bar{x})$.

Proof. The same as for Theorem 3.

Let

$y<x \Leftrightarrow \exists z(\neg z=0\wedge z+y=x);$

$\forall y<x\varphi \Leftrightarrow \forall y(y<x\supset\varphi).$

Lemma 10. *Let* $\mathbf{HA}\vdash\forall x(\varphi(x)\vee\neg\varphi(x))$. *Then* $\mathbf{HA}\vdash\forall x(\forall y<x\varphi(y)\vee\neg\forall y<x\varphi(y))$.

Proof. Immediate.

Theorem 11. *Let* R *be a decidable set of natural numbers. Then there exists a pair of formulae* $\mathrm{In}(\bar{x}), P(\bar{x})$ *such that* $\mathrm{In}(\bar{x})$ *is a Harrop formula,* $\mathrm{In}(\bar{x}), P(\bar{x})$ *specify* R *on* $\mathbf{N}$ *and* $\mathbf{HA}+\mathbf{MP}\vdash\forall\bar{x}(\mathrm{In}(\bar{x})\supset(P(\bar{x})\vee\neg P(\bar{x})))$.

Proof. Let $f(\bar{x})$ be the total recursive function such that $f(\bar{x})=0$ iff $\bar{x}\in R$. Let $F(\bar{x},\bar{y},z)$ be a quantifier-free formula of the arithmetical language such that for any tuple $\bar{m},n$ of natural numbers $f(\bar{m})=n$ iff $\exists\bar{y}(F(\underline{\bar{m}},\bar{y},\underline{n})$. Let $G(\bar{x},u)\Leftrightarrow\exists\bar{y}z(F(\bar{x},\bar{y},z)\wedge C_n(\bar{y},z,u))$ and $H(\bar{x},u)\Leftrightarrow G(\bar{x},u)\wedge(\forall v<u)\neg G(\bar{x},u)$. In Theorem 6 we proved that $\mathbf{HA}\vdash\forall\bar{x}u(G(\bar{x},u)\vee\neg G(\bar{x},u))$. From this using Lemma 10 we obtain $\mathbf{HA}\vdash\forall\bar{x}u(H(\bar{x},u)\vee\neg H(\bar{x},u))$. We let $\mathrm{In}(\bar{x})\Leftrightarrow\neg\neg\exists uH(\bar{x},u)$. Since f is totally defined, $\forall\bar{x}\mathrm{In}(\bar{x})$ is true. Applying the Markov's principle we have $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\bar{x})\vdash\exists uH(\bar{x},u)$. Let $P(\bar{x})\Leftrightarrow\exists u\bar{y}(H(\bar{x},u)\wedge C_n(\bar{y},0,u))$. First we prove that the pair $\mathrm{In}(\bar{x}), P(\bar{x})$ specifies R on $\mathbf{N}$. It follows from the fact that $\forall\bar{x}\mathrm{In}(\bar{x})$ is true and from the following equivalencies:

$\bar{x}\in R\Leftrightarrow$
$f(\bar{x})=0\Leftrightarrow$
$\exists\bar{y}(F(\bar{x},\bar{y},0)\Leftrightarrow$
$\exists u\bar{y}(H(\bar{x},u)\wedge C_n(\bar{y},0,u))\Leftrightarrow$
$P(\bar{x})$.

From the definition of H it is possible to prove that $\mathbf{HA}\vdash\forall\bar{x}u_1u_2(H(\bar{x},u_1)\wedge H(\bar{x},u_2)\supset u_1=u_2)$. Using it we obtain $\mathbf{HA}\vdash\forall\bar{x}\bar{y}uv(H(\bar{x},u)\wedge C_n(\bar{y},v,u)\wedge\neg v=0\supset\neg P(\bar{x}))$. Finally, using it, $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\bar{x})\vdash\exists uH(\bar{x},u)$, properties of C_n and $\mathbf{HA}\vdash\forall v(v=0\vee\neg v=0)$ we can obtain $\mathbf{HA}+\mathbf{MP}+\mathrm{In}(\bar{x})\vdash P(\bar{x})\vee\neg P(\bar{x})$ or $\mathbf{HA}+\mathbf{MP}\vdash\forall\bar{x}(\mathrm{In}(\bar{x})\supset(P(\bar{x})\vee\neg P(\bar{x})))$.

The last two theorems essentially mean that we can extract all decidable predicates from proofs in $\mathbf{HA}+\mathbf{MP}$ but in general it is impossible to prove that such a predicate is defined on the set of all natural numbers.

References

1. R Harrop. Concerning formulas of the types $A \to B \wedge C$, $A \to (Ex)B(x)$ in intuitionistic formal system. *J. of Symb. Logic*, 17:27–32, 1960.
2. S.C. Kleene. Disjunction and existence under implication in elementary intuitionistic formalism. *Journal of Symbolic Logic*, 27(1), 1962.
3. Matiyasevič. The diophantiness of recursively enumerable sets (in Russian). *Soviet Mathematical Doklady*, pages 279–282, 1970.