Some Subrecursive Versions of Grzegorczyk's Uniformity Theorem

Dimiter Skordev*

Faculty of Mathematics and Computer Science, University of Sofia
blvd. James Bourchier 5, Sofia 1164, Bulgaria

Key words Uniformity Theorem, computable functional, primitive recursive functional, elementary recursive functional, computable analysis.
MSC (2000) 03D99, 03F60

A theorem published by A. Grzegorczyk in 1955 states a certain kind of effective uniform continuity of computable functionals that transform systems of total functions in the set of the natural numbers and of natural numbers into natural numbers. Namely, for any such functional a computable functional with one function-argument and the same number-arguments exists such that the values of the first of the functionals at functions dominated by a given one are completely determined by their restrictions to numbers not exceeding the corresponding values of the second functional at the given function. We prove versions of this theorem for the class of the primitive recursive functionals and for the class of the elementary recursive ones. Analogous results can be proved for many other subrecursive classes of functionals.

1 Introduction

Let $\mathbb{N}$ be the set of the natural numbers, and $\mathcal{F}$ be the set of all total functions from $\mathbb{N}$ into $\mathbb{N}$. For any natural numbers n and k the mappings of $\mathcal{F}^n \times \mathbb{N}^k$ into $\mathbb{N}$ will be called *functionals with n function-arguments and k number-arguments* (abbreviations $\bar{\boldsymbol{f}}$ and $\bar{\boldsymbol{x}}$ will be used for function-arguments $f_1, \ldots, f_n$ and number-arguments $x_1, \ldots, x_k$ of such a functional; the same sort of abbreviations with other characters instead of f and x will be also used). For any f and g in $\mathcal{F}$, the inequality $g \le f$ will mean that $g(x) \le f(x)$ for all x in $\mathbb{N}$. For any f in $\mathcal{F}$, let $\mathcal{F}_f$ be the set of all g in $\mathcal{F}$ satisfying the inequality $g \le f$, and for any x in $\mathbb{N}$, let $\mathbb{N}_x$ be the finite set $\{0, 1, \ldots, x\}$. The following property of computable functionals is proved in [1] under the name "Uniformity Theorem", and an application of it to computable real functions is given there. (A short and transparent proof of this property is given in [4].)

If Φ is a computable functional with n function-arguments and k number-arguments then a computable functional Ω with one function-argument and k number-arguments exists such that for any $x_1, \ldots, x_k$ in $\mathbb{N}$, any f in $\mathcal{F}$ and any $g_1, \ldots, g_n, h_1, \ldots, h_n$ in $\mathcal{F}_f$ the equalities $g_i{\restriction}\mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})} = h_i{\restriction}\mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})}$, $i = 1, \ldots, n$, (with $\restriction$ meaning "restricted to") imply the equality $\Phi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}) = \Phi(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}})$.

In its form from [1, 4] the above property can be applied to computable analysis with no restriction imposed on the complexity of transformation of the arguments into the function values. On the other hand, the concrete functions encountered in analysis have usually such a transformation complexity that is low from the point of view of recursive function theory. Therefore one could be interested in subrecursive versions of the property that concern certain proper subclasses of the class of all computable functionals. We propose here two such versions (with proofs that are more constructive in a certain sense than the proof of the original property).

* e-mail: skordev@fmi.uni-sofia.bg

2 A primitive recursive version of the Uniformity Theorem

A primitive recursive version of the Uniformity Theorem can be obtained from the original one by replacing the word "computable" with the words "primitive recursive" (i.e. for any primitive recursive Φ the existence of a corresponding primitive recursive Ω is asserted). Of course, the definition of the notion of a primitive recursive functional is by induction. We shall use this definition in the following form.

1. The functional $\lambda f_1 \dots f_n \,.\, 0$ is primitive recursive.
2. The functional $\lambda f_1 \dots f_n x \,.\, x+1$ is primitive recursive.
3. The functionals $\lambda f_1 \dots f_n x_1 \dots x_k \,.\, x_j$, $j = 1, \dots, k$, are primitive recursive.
4. The functionals $\lambda f_1 \dots f_n x \,.\, f_i(x)$, $i = 1, \dots, n$, are primitive recursive.
5. If Φ_0 is a primitive recursive functional with n function-arguments and m number-arguments, and $\Phi_1, \dots, \Phi_m$ are primitive recursive functionals with n function-arguments and k number-arguments, then the functional
$$\lambda f_1 \dots f_n x_1 \dots x_k \,.\, \Phi_0(\bar{\boldsymbol{f}}, \Phi_1(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}), \dots, \Phi_m(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}))$$
is also primitive recursive.
6. If Φ_0 is a primitive recursive functional with n function-arguments and k number-arguments, Φ_1 is a primitive recursive functional with n function-arguments and $k+2$ number-arguments, and the functional Φ is defined by means of the equalities
$$\begin{aligned} \Phi(\bar{\boldsymbol{f}}, 0, \bar{\boldsymbol{x}}) &= \Phi_0(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})\,, \\ \Phi(\bar{\boldsymbol{f}}, t+1, \bar{\boldsymbol{x}},) &= \Phi_1(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}}, \Phi(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}}))\,, \end{aligned}$$
then Φ is also primitive recursive.

A functional Ψ with n function-arguments and k number arguments will be called *monotonically increasing* if $\Psi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \le \Psi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{y}})$ whenever $f_1, \dots, f_n, g_1, \dots, g_n$ belong to $\mathcal{F}$, $x_1, \dots, x_k, y_1, \dots, y_k$ belong to $\mathbb{N}$ and the inequalities $f_1 \le g_1, \ \dots, \ f_n \le g_n, \ x_1 \le y_1, \ \dots, \ x_n \le y_n$ hold. We shall first prove the following statement.

Lemma 2.1 *Let Φ be a primitive recursive functional with n function-arguments and k number-arguments. Then there is a monotonically increasing primitive recursive functional Ψ with the same number of function- and number-arguments such that $\Phi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \le \Psi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})$ for all $f_1, \dots, f_n$ in $\mathcal{F}$ and all $x_1, \dots, x_k$ in $\mathbb{N}$.*

Proof. If Φ is primitive recursive according to some of the first three clauses of the definition then we set $\Psi = \Phi$, and if $\Phi = \lambda f_1 \dots f_m x \,.\, f_i(x)$ then we set
$$\Psi(\bar{\boldsymbol{f}}, x) = \max\{f_i(t) | t \in \mathbb{N}_x\}.$$
If $\Phi = \lambda f_1 \dots f_n x_1 \dots x_k \,.\, \Phi_0(\bar{\boldsymbol{f}}, \Phi_1(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}), \dots, \Phi_m(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}))$, where the functionals $\Phi_0, \Phi_1, \dots, \Phi_m$ are dominated by the monotonically increasing primitive recursive functionals $\Psi_0, \Psi_1, \dots, \Psi_m$, respectively, then we set
$$\Psi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) = \Psi_0(\bar{\boldsymbol{f}}, \Psi_1(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}), \dots, \Psi_m(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})).$$
Suppose now Φ is defined by means of the equalities in clause 6, and let Φ_0 and Φ_1 be dominated by the monotonically increasing primitive recursive functionals Ψ_0 and Ψ_1, rerspectively. Then we define Ψ by means of the equalities
$$\begin{aligned} \Psi(\bar{\boldsymbol{f}}, 0, \bar{\boldsymbol{x}}) &= \Psi_0(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})\,, \\ \Psi(\bar{\boldsymbol{f}}, t+1, \bar{\boldsymbol{x}}) &= \max\{\Psi_1(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}}, \Psi(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}})), \Psi_0(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})\}\,. \end{aligned}$$

The functional Ψ is also primitive recursive. Its monotonicity and the inequality $\Phi(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}}) \leq \Psi(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}})$ can be easily verified by induction. □

Corollary 2.2 *If Φ is a primitive recursive functional with n function-arguments and k number-arguments then a monotonically increasing primitive recursive functional Θ with one function-argument and k number-arguments exists such that for any $x_1, \ldots, x_k$ in $\mathbb{N}$, any f in $\mathcal{F}$ and any $f_1, \ldots, f_n$ in $\mathcal{F}_f$ the inequality $\Phi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \leq \Theta(f, \bar{\boldsymbol{x}})$ holds.*

Proof. Let Φ be a primitive recursive functional with n function-arguments, and let Ψ be a monotonically increasing primitive recursive functional with the same number function- and number-arguments that dominates Φ. For any $x_1, \ldots, x_k$ in $\mathbb{N}$, any f in $\mathcal{F}$ and any $f_1, \ldots, f_n$ in $\mathcal{F}_f$ we have the inequalities

$$\Phi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \leq \Psi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \leq \Psi(f, \ldots, f, \bar{\boldsymbol{x}}).$$

Thus we may set $\Theta(f, \bar{\boldsymbol{x}}) = \Psi(f, \ldots, f, \bar{\boldsymbol{x}})$. □

Remark 2.3 The above corollary can also be proved directly by means of an induction similar to the one in the proof of Lemma 2.1.

Making use of Corollary 2.2, we shall prove the following statement that slightly strengthens the promised primitive recursive version of Grzegorczyk's Uniformity Theorem (the strengthening concerns the monotonicity of Ω).

Theorem 2.4 *If Φ is a primitive recursive functional with n function-arguments and k number-arguments then a monotonically increasing primitive recursive functional Ω with one function-argument and k number-arguments exists such that for any $x_1, \ldots, x_k$ in $\mathbb{N}$, any f in $\mathcal{F}$ and any $g_1, \ldots, g_n, h_1, \ldots, h_n$ in $\mathcal{F}_f$ the equalities $g_i \restriction \mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})} = h_i \restriction \mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})}$, $i = 1, \ldots, n$, imply the equality $\Phi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}) = \Phi(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}})$.*

Proof. If Φ is primitive recursive according to some of the first three clauses of the definition then we set $\Omega(f, \bar{\boldsymbol{x}}) = 0$, and if $\Phi = \lambda f_1 \ldots f_n x \,.\, f_i(x)$ then we set $\Omega(f, x) = x$.

Suppose now that

$$\Phi = \lambda f_1 \ldots f_n x_1 \ldots x_k \,.\, \Phi_0(\bar{\boldsymbol{f}}, \Phi_1(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}), \ldots, \Phi_m(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})),$$

where $\Phi_0, \Phi_1, \ldots, \Phi_m$ are primitive recursive functionals with corresponding monotonically increasing primitive recursive functionals $\Omega_0, \Omega_1, \ldots, \Omega_m$ having the needed properties. We set

$$\Omega(f, \bar{\boldsymbol{x}}) = \max\{\Omega_0(f, \Theta_1(f, \bar{\boldsymbol{x}}), \ldots, \Theta_m(f, \bar{\boldsymbol{x}})), \Omega_1(f, \bar{\boldsymbol{x}}), \ldots, \Omega_m(f, \bar{\boldsymbol{x}})\},$$

where $\Theta_1, \ldots, \Theta_m$ are monotonically increasing primitive recursive functionals such that

$$\Phi_l(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \leq \Theta_l(f, \bar{\boldsymbol{x}}),\ l = 1, \ldots, m,$$

whenever $f_1, \ldots, f_n$ belong to $\mathcal{F}_f$. The functional Ω is primitive recursive and monotonically increasing. Let $x_1, \ldots, x_k$ belong to $\mathbb{N}$, f be in $\mathcal{F}$ and $g_1, \ldots, g_n, h_1, \ldots, h_n$ be in $\mathcal{F}_f$. Let also the equalities $g_i \restriction \mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})} = h_i \restriction \mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})}$, $i = 1, \ldots, n$, hold. Then

$$\begin{aligned} \Phi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}) = \Phi_0(\bar{\boldsymbol{g}}, \Phi_1(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}), \ldots, \Phi_m(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}})) &= \Phi_0(\bar{\boldsymbol{h}}, \Phi_1(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}), \ldots, \Phi_m(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}})) \\ &= \Phi_0(\bar{\boldsymbol{h}}, \Phi_1(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}}), \ldots, \Phi_m(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}})) = \Phi(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}}), \end{aligned}$$

since $\Omega_0(f, \Phi_1(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}), \ldots, \Phi_m(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}})) \leq \Omega_0(f, \Theta_1(f, \bar{\boldsymbol{x}}), \ldots, \Theta_m(f, \bar{\boldsymbol{x}})) \leq \Omega(f, \bar{\boldsymbol{x}})$, and $\Omega_l(f, \bar{\boldsymbol{x}}) \leq \Omega(f, \bar{\boldsymbol{x}})$ for $l = 1, \ldots, m$.

Finally suppose the functional Φ is defined by means of the equalities from clause 6, where Φ_0 and Φ_1 are primitive recursive functionals with corresponding monotonically increasing primitive recursive functionals Ω_0 and Ω_1 having the needed properties. By the primitive recursiveness of Φ, there is also a monotonically increasing primitive recursive functional Θ such that $\Phi(\bar{\boldsymbol{f}}, t, \bar{\boldsymbol{x}}) \leq \Theta(f, t, \bar{\boldsymbol{x}})$ whenever $f_1, \ldots, f_n$ belong to $\mathcal{F}_f$. Let us define a functional Ω by means of the following equalities:

$$\begin{aligned} \Omega(f, 0, \bar{\boldsymbol{x}}) &= \Omega_0(f, \bar{\boldsymbol{x}}), \\ \Omega(f, t+1, \bar{\boldsymbol{x}}) &= \max\{\Omega_1(f, t, \bar{\boldsymbol{x}}, \Theta(f, t, \bar{\boldsymbol{x}})), \Omega_0(f, \bar{\boldsymbol{x}})\}. \end{aligned}$$

The functional Ω is monotonically increasing and primitive recursive too. Let $x_1, \ldots, x_k$ belong to $\mathbb{N}$, f be in $\mathcal{F}$ and $g_1, \ldots, g_n, h_1, \ldots, h_n$ be in $\mathcal{F}_f$. We shall prove by induction on t the implication

$$g_i \restriction \mathbb{N}_{\Omega(f,t,\bar{\boldsymbol{x}})} = h_i \restriction \mathbb{N}_{\Omega(f,t,\bar{\boldsymbol{x}})} \text{ for } i = 1, \ldots, n \ \Rightarrow\ \Phi(\bar{\boldsymbol{g}}, t, \bar{\boldsymbol{x}}) = \Phi(\bar{\boldsymbol{h}}, t, \bar{\boldsymbol{x}}).$$

In the case of $t = 0$ this implication is true, since it reduces then to the implication

$$g_i \restriction \mathbb{N}_{\Omega_0(f,\bar{\boldsymbol{x}})} = h_i \restriction \mathbb{N}_{\Omega_0(f,\bar{\boldsymbol{x}})} \text{ for } i = 1, \ldots, n \ \Rightarrow\ \Phi_0(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}) = \Phi_0(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}}).$$

On the other hand, if for a given natural number t the considered implication is true together with the equalities $g_i \restriction \mathbb{N}_{\Omega(f,t+1,\bar{\boldsymbol{x}})} = h_i \restriction \mathbb{N}_{\Omega(f,t+1,\bar{\boldsymbol{x}})}$, $i = 1, \ldots, n$, then

$$\begin{aligned} \Phi(\bar{\boldsymbol{g}}, t+1, \bar{\boldsymbol{x}}) = \Phi_1(\bar{\boldsymbol{g}}, t, \bar{\boldsymbol{x}}, \Phi(\bar{\boldsymbol{g}}, t, \bar{\boldsymbol{x}})) &= \Phi_1(\bar{\boldsymbol{h}}, t, \bar{\boldsymbol{x}}, \Phi(\bar{\boldsymbol{g}}, t, \bar{\boldsymbol{x}})) \\ &= \Phi_1(\bar{\boldsymbol{h}}, t, \bar{\boldsymbol{x}}, \Phi(\bar{\boldsymbol{h}}, t, \bar{\boldsymbol{x}})) = \Phi(\bar{\boldsymbol{h}}, t+1, \bar{\boldsymbol{x}}) \end{aligned}$$

due to the inequalities $\Omega_1(f, t, \bar{\boldsymbol{x}}, \Phi(\bar{\boldsymbol{g}}, t, \bar{\boldsymbol{x}})) \leq \Omega_1(f, t, \bar{\boldsymbol{x}}, \Theta(f, t, \bar{\boldsymbol{x}})) \leq \Omega(f, t+1, \bar{\boldsymbol{x}})$ and the inequality $\Omega(f, t, \bar{\boldsymbol{x}}) \leq \Omega(f, t+1, \bar{\boldsymbol{x}})$. □

3 An elementary recursive version of the Uniformity Theorem

The notion of an elementary recursive functional is introduced in [1] by using bounded primitive recursion. An equivalent definition in the style of [2] (cf. also [3], pp. 285–286) can be given to this notion. Here is a definition in that style.

1. The functional $\lambda f_1 \ldots f_n . 1$ is elementary recursive.
2. The functional $\lambda f_1 \ldots f_n xy . x \dot{-} y$ is elementary recursive.
3. The functionals $\lambda f_1 \ldots f_n x_1 \ldots x_k . x_j$, $j = 1, \ldots, k$, are elementary recursive.
4. The functionals $\lambda f_1 \ldots f_n x . f_i(x)$, $i = 1, \ldots, n$, are elementary recursive.
5. If Φ_0 is a elementary recursive functional with n function-arguments and m number-arguments, and $\Phi_1, \ldots, \Phi_m$ are elementary recursive functionals with n function-arguments and k number-arguments, then the functional

$$\lambda f_1 \ldots f_n x_1 \ldots x_k . \Phi_0(\bar{\boldsymbol{f}}, \Phi_1(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}), \ldots, \Phi_m(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}))$$

   is also elementary recursive.
6. If Φ_0 is a elementary recursive functional with n function-arguments and $k+1$ number-arguments, then the functionals

$$\lambda f_1 \ldots f_n t x_1 \ldots x_k . \sum_{s=0}^{t} \Phi_0(\bar{\boldsymbol{f}}, s, \bar{\boldsymbol{x}}), \quad \lambda f_1 \ldots f_n t x_1 \ldots x_k . \prod_{s=0}^{t} \Phi_0(\bar{\boldsymbol{f}}, s, \bar{\boldsymbol{x}})$$

   are also elementary recursive.

An elementary recursive version of the Uniformity Theorem can be obtained from the original one by replacing the word "computable"with the words "elementary recursive" (i.e. for any elementary recursive Φ the existence of a corresponding elementary recursive Ω is asserted). The proof is quite similar to the proof of the primitive recursive version, therefore we shall describe it briefly, mentioning only some changes that do not simply reduce to replacement of "primitive" by "elementary".

Lemma 3.1 *Let Φ be an elementary recursive functional with n function-arguments and k number-arguments. Then there is a monotonically increasing elementary recursive functional Ψ with the same number of function- and number-arguments such that $\Phi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \leq \Psi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}})$ for all $f_1, \ldots, f_n$ in $\mathcal{F}$ and all $x_1, \ldots, x_k$ in $\mathbb{N}$.*

Proof. The construction of Ψ is the same as in Lemma 2.1, except for the case of the second and of the last clause of the definition. If $\Phi = \lambda f_1 \ldots f_n xy \,.\, x \dot{-} y$ then we set $\Psi = \lambda f_1 \ldots f_n xy \,.\, x$. If

$$\Phi = \lambda f_1 \ldots f_n t x_1 \ldots x_k \,.\, \sum_{s=0}^{t} \Phi_0(\bar{\boldsymbol{f}}, s, \bar{\boldsymbol{x}})\,,$$

and Ψ_0 is a monotonically increasing elementary recursive functional dominating Φ_0, then we set

$$\Psi = \lambda f_1 \ldots f_n t x_1 \ldots x_k \,.\, \sum_{s=0}^{t} \Psi_0(\bar{\boldsymbol{f}}, s, \bar{\boldsymbol{x}})\,,$$

and similarly in the case of product instead of sum. □

Corollary 3.2 *If Φ is an elementary recursive functional with n function-arguments and k number-arguments then a monotonically increasing elementary recursive functional Θ with one function-argument and k number-arguments exists such that for any $x_1, \ldots, x_k$ in $\mathbb{N}$, any f in $\mathcal{F}$ and any $f_1, \ldots, f_n$ in $\mathcal{F}_f$ the inequality $\Phi(\bar{\boldsymbol{f}}, \bar{\boldsymbol{x}}) \leq \Theta(f, \bar{\boldsymbol{x}})$ holds.*

Proof. In the same way as the proof of Corollary 2.2. □

Theorem 3.3 *If Φ is an elementary recursive functional with n function-arguments and k number-arguments then a monotonically increasing elementary recursive functional Ω with one function-argument and k number-arguments exists such that for any $x_1, \ldots, x_k$ in $\mathbb{N}$, any f in $\mathcal{F}$ and any $g_1, \ldots, g_n, h_1, \ldots, h_n$ in $\mathcal{F}_f$ the equalities $g_i{\restriction}\mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})} = h_i{\restriction}\mathbb{N}_{\Omega(f,\bar{\boldsymbol{x}})}$, $i = 1, \ldots, n$, imply the equality $\Phi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{x}}) = \Phi(\bar{\boldsymbol{h}}, \bar{\boldsymbol{x}})$.*

Proof. Only one divergence from the proof of Theorem 2.4 needs to be mentioned, and it leads to a simplification. Namely, if Φ is elementary recursive according to clause 6, and Ω_0 is a monotonically increasing elementary recursive functional having the needed property with respect to Φ_0, then we set $\Omega = \Omega_0$ (both in the case when Φ is defined by means of summation and in the case when it is defined by means of a product). □

4 Concluding remark

It is quite clear that analogous results can be proved in a similar way for many other subrecursive classes of functionals. Perhaps all these versions of the Uniformity Theorem can be nicely captured by some appropriate general formulation.

5 Acknowledgments

Thanks are due to the anonymous referees for their useful suggestions.

References

[1] Grzegorczyk, A., Computable functionals. Fund. Math., 42, (1955), 168–202.
[2] Kalmár, L., A simple example of an unsolvable arithmetical problem. Matematikai és Fizikai Lapok, 50, (1943) 1–23 (in Hungarian).
[3] Kleene, S. C., Introduction to Metamathematics. New York–Toronto, 1952.
[4] Kleene, S. C., A note on computable functionals. Nederl. Acad. Wet., Proc. Ser. A, 59, (1956), 275–280.