Synthesis of moduli of uniform continuity by the Monotone Dialectica Interpretation in the proof-system MinLog

Mircea-Dan Hernest[1,2]

Laboratoire d'Informatique (LIX)
École Polytechnique
F-91128 Palaiseau - FRANCE

Abstract

We extract on the computer a number of moduli of uniform continuity for the first few elements of a sequence of closed terms t of Gödel's $\mathbf{T}$ of type $(\mathbb{N}\to\mathbb{N})\to(\mathbb{N}\to\mathbb{N})$. The generic solution may then be quickly inferred by the human. The automated synthesis of such moduli proceeds from a proof of the hereditarily extensional equality ($\approx$) of t to itself, hence a proof in a weakly extensional variant of Berger-Buchholz-Schwichtenberg's system Z of $t \approx_{(\mathbb{N}\to\mathbb{N})\to(\mathbb{N}\to\mathbb{N})} t$. We use an implementation on the machine, in Schwichtenberg's `MinLog` proof-system, of a non-literal adaptation to Natural Deduction of Kohlenbach's monotone functional interpretation. This new version of the Monotone Dialectica produces terms in `NbE`-normal form by means of a recurrent partial `NbE`-normalization. Such partial evaluation is strictly necessary.

Keywords: Program extraction from (classical) proofs, Complexity of extracted programs, Proof- and program-extraction system `MinLog`, Gödel's functional interpretation, Partial Evaluation, Proof Mining, Monotone Dialectica Interpretation.

1 The *monotone* functional *Dialectica* interpretation

Kohlenbach's monotone variant of Gödel's functional (aka "Dialectica") interpretation was introduced in [18] as an optimization of Gödel's original term extraction technique [3] from [8]. The main feature of this "monotone Dialectica interpretation" is the extraction of Howard majorants [14] (or, equally, Bezem strong majorants [6]) [4] for some exact realizers [5]. In the mathematical practice this operation turns out to be much simpler [6] than the synthesis of some actual exact realizers by the pure Gödel's Dialectica interpretation from [8,1].

---

[1] *Project LogiCal* - Pôle Commun de Recherche en Informatique du Plateau de Saclay, CNRS, École Polytechnique, INRIA et Université Paris-Sud - FRANCE

[2] Email: `danher@lix.polytechnique.fr`

[3] Paper [1] provides a nice survey in English which includes the extensions to full Analysis.

[4] In this paper we actively use only Howard's variant of majorization, originally introduced in [14] (see also [20,21]), which is presented in Definition 1.3 below.

[5] Which are not effectively produced, but their *strong* existence is ensured intuitionistically.

[6] See, e.g., [21] and [22] for two comprehensive surveys of the wide range of mathematical application of this purely proof-theoretical technique.

This paper is electronically published in
Electronic Notes in Theoretical Computer Science
URL: `www.elsevier.nl/locate/entcs`

Definition 1.1 [Base Arithmetic for Monotone Dialectica program-extraction]
We denote by $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$ the weakly extensional variant (see [10]) of Berger-Buchholz-Schwichtenberg's system Z (introduced in [2], see also [24]) to which the *strong* $\exists$ quantifier was added (together with its defining axioms, see [10,24]) and also all the necessary *monotonic* elements were added, namely the functional inequality constant $\geq$ together with the axioms governing its usual behaviour [7].

Note that the system $\mathtt{WeZ}^{\exists}$, i.e., $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$ without the monotonic elements (which was denoted $\mathtt{WE{-}Z}^{-}$ in [10]) is a Natural Deduction formulation of the weakly extensional Heyting Arithmetic in all finite types $\mathtt{WE{-}HA}^{\omega}$ from Section 1.6.12 of [26].

Definition 1.2 [Extended Arithmetic for extraction by Monotone Dialectica]
We denote by $\mathtt{WeZ}^{\exists+}_{\mathtt{m}}$ the extension of $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$ with the Independence of Premises for universal premises, the Axiom of Choice and Markov's Principle (axiom) [8].

Definition 1.3 [Section 2 of [14], adapted to the **T** presentation from [10] [9]]
Howard's majorizability relation $\succeq$ is defined over the **T** type structure by

$$
\begin{aligned}
x \succeq_{\mathbb{N}} y \;&:\equiv\; \mathtt{at}(\geq x y) \\
x \succeq_{\rho\tau} y \;&:\equiv\; \forall z_1^{\rho}, z_2^{\rho}\,(z_1 \succeq_{\rho} z_2 \rightarrow x z_1 \succeq_{\tau} y z_2) \quad ,
\end{aligned}
$$

where $\geq$ is the usual inequality boolean function on $\mathbb{N}\times\mathbb{N}$ defined in [10] and "at" is the boolean, unary and unique predicate of $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$, also defined in [10].

The monotone Dialectica interpretation (abbreviated "MD-interpretation" and even shorter, MDI) is a recursive syntactic translation from proofs in $\mathtt{WeZ}^{\exists+}_{\mathtt{m}}$ [10] to proofs in $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$ such that the positive occurrences of the strong $\exists$ and the negative occurrences of $\forall$ in the proof's conclusion formula get effectively (either Howard or Bezem) majorized at each of the proof-recursion steps [11] by terms in Gödel's **T**. These *majorizing terms* are also called "the programs extracted by" the MDI and (if only the extracted terms are wanted) this translation process is also referred to as "Monotone Dialectica program-extraction".

Definition 1.4 [*Association of boolean terms to quantifier-free formulas*]
By quantifier-free formula we understand a formula built from prime formulas $\mathtt{at}(t^{bool})$ and $\perp$ by means of $\wedge$, $\rightarrow$ and, if $\exists$ is available, also $\vee$. Such formulas are decidable in $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$. There exists a unique bijective association of boolean terms to quantifier-free formulas $A_0 \mapsto \mathtt{t}_{A_0}$ such that $\mathtt{WeZ}^{\exists}_{\mathtt{m}} \vdash A_0 \leftrightarrow \mathtt{at}(\mathtt{t}_{A_0})$.

The MD-interpretation of proofs includes the following translation of formulas:

---

[7] See Section 3.1 of [10] for details - our system $\mathtt{WeZ}^{\exists}_{\mathtt{m}}$ here was there denoted by $\mathtt{WE{-}Z}^{-}_{\mathtt{m}}$.
[8] See, e.g., Section 2.3 of [10] for the detailed definitions of these axioms (plus comments).
[9] Please beware of the typo in the corresponding definition from Section 3.1 of [10].
[10] This can be extended to fully classical proofs, modulo some double-negation translation.
[11] This is exactly *the point* of Kohlenbach's MD-interpretation from [18], in contrast to his precursor of the MDI from [16] which first extracts the effective Gödel's Dialectica *exact realizers* and subsequently majorizes them via the algorithms of either Howard [14] or Bezem [6].

Definition 1.5 [**The MD-interpretation of formulas**] Recursively defined:

$$A^{\mathtt{MD}} :\equiv A_{\mathtt{MD}} :\equiv \mathtt{at}(\mathtt{t}_A) \text{ for quantifier-free formulas } A$$

$$(A \wedge B)^{\mathtt{MD}} :\equiv \exists \underline{x}, \underline{u} \forall \underline{y}, \underline{v} \, [\, (A \wedge B)_{\mathtt{MD}} :\equiv A_{\mathtt{MD}}(\underline{x}; \underline{y}; \underline{a}) \wedge B_{\mathtt{MD}}(\underline{u}; \underline{v}; \underline{b}) \,]$$

$$(\exists z A(z, \underline{a}))^{\mathtt{MD}} :\equiv \exists z^{\dagger}, \underline{x} \forall \underline{y} \, [\, (\exists z A(z, \underline{a}))_{\mathtt{MD}}(z^{\dagger}, \underline{x}; \underline{y}; \underline{a}) :\equiv A_{\mathtt{MD}}(\underline{x}; \underline{y}; z^{\dagger}, \underline{a}) \,]$$

$$(\forall z A(z, \underline{a}))^{\mathtt{MD}} :\equiv \exists \underline{X} \forall z^{\dagger}, \underline{y} \, [\, (\forall z A(z, \underline{a}))_{\mathtt{MD}}(\underline{X}; z^{\dagger}, \underline{y}; \underline{a}) :\equiv A_{\mathtt{MD}}(\underline{X}(z^{\dagger}); \underline{y}; z^{\dagger}, \underline{a}) \,]$$

$$(A \to B)^{\mathtt{MD}} :\equiv \exists \underline{Y}, \underline{U} \forall \underline{x}, \underline{v} \, [\, (A \to B)_{\mathtt{MD}} :\equiv A_{\mathtt{MD}}(\underline{x}; \underline{Y}(\underline{x}, \underline{v})) \to B_{\mathtt{MD}}(\underline{U}(\underline{x}); \underline{v}) \,]$$

where $\cdot \mapsto \cdot^{\dagger}$ is a mapping which assigns to every given variable z a completely new variable $z^{\dagger}$ which has the same type of z. The free variables of $A^{\mathtt{MD}}$ are exactly the free variables of A.

Theorem 1.6 (Majorant realizer synthesis by the MD-interpretation) [12]
There exists an algorithm which, given at input a Natural Deduction proof $\mathcal{P} : \{C^i(a_i)\}_{i=1}^n \vdash A(a)$ [hence of the conclusion formula A, whose free variables form the *tuple* a, from the *undischarged* assumption formulas $\{C^i\}_{i=1}^n$] in $\mathtt{WeZ}_{\mathtt{m}}^{\exists +}$, it produces at output the following (below let $\underline{a} :\equiv a_1, \ldots, a_n, a$):

(i) the tuples of terms $\{T_i[\underline{a}]\}_{i=1}^n$ and $T[\underline{a}]$, whose free variables are among $\underline{a}$
(ii) the tuples of variables $\{x_i\}_{i=1}^n$ and y, all together with
(iii) the following verifying proof in $\mathtt{WeZ}_{\mathtt{m}}^{\exists}$ (below let $\underline{x} :\equiv x_1, \ldots, x_n$):

$$\mathcal{P}_{\mathtt{MD}} : \emptyset \vdash \exists Y_1, \ldots Y_n, X \, [\, \textstyle\bigwedge_{i=1}^n (\lambda \underline{a}.\, T_i) \succeq Y_i \wedge (\lambda \underline{a}.\, T) \succeq X \wedge$$
$$\forall \underline{a}, \underline{x}, y \, (\, \{ \textstyle\bigwedge_{i=1}^n C^i_{\mathtt{MD}}(x_i; Y_i(\underline{a}, \underline{x}, y); a_i) \} \to A_{\mathtt{MD}}(X(\underline{a}, \underline{x}); y; a) \,) \,]$$

Moreover, variables $\underline{x}$ and y do not occur in $\mathcal{P}$ (they are all completely new). Hence $\underline{x}$ and y also do not occur free in the *extracted* terms $\{T_i\}_{i=1}^n$ and T.

Proof: See [11] for a sketch of the proof (in Natural Deduction) or [18,21] for full proofs of the equivalent original formulations in the Hilbert-style setting. □

Remark 1.7 The MD-translated proof $\mathcal{P}_{\mathtt{MD}}$ is also called the *verifying proof* since it arithmetically verifies the fact that the MD-extracted programs actually *majorize* some (strong, intuitionistically proven to exist) realizers of the MD-interpretation of the conclusion formula of the proof at input.

Gödel's Dialectica interpretation becomes far more complicated when it has to face Contraction, which in Natural Deduction amounts to the discharging of more than one copy of an uncancelled assumption in an Implication Introduction $\dfrac{[A] \ldots /B}{A \to B}$. This is because, for the contractions which are relevant to Dialectica [13],

[12] This theorem was conjectured (in a weaker form) already in Section 3.1 of [10].
[13] Not all *logical* contractions are relevant for the Dialectica interpretations, see [12] for a short account of this issue or [11] for full details.

the contraction formula A becomes [14] part of the raw (not yet normalized) realizing term. A number of such *D-relevant* contraction formulas, which would not be part of the executed finally strongly normalized extracted term, can be eliminated already at the extraction stage, see [12] for such an example. Unfortunately, such an a priori elimination during extraction of some of the contractions (which we named "redundant" in [12]) is not always possible, see also [12] for such a negative example. The MD-interpretation simplifies the Dialectica treatment of all non-redundant relevant contractions and therefore represents an important complexity improvement of the extracted program whenever such "persistent" contractions occur in the proof at input.

2 The minimal arithmetic HeExtEq proof in MinLog

MinLog is an interactive proof- and program-extraction system developed by H. Schwichtenberg and members of the logic group at the University of Munich. It is based on first order Natural Deduction calculus and uses as primitive *minimal* rather than classical or intuitionistic logic. See [9,25] for full details.

The hereditarily-extensional-equality test-case (abbreviated HeExtEq) was suggested by U. Kohlenbach as an interesting example for the application of the Monotone Dialectica program extraction from proofs, see Chapter 8 of [21]. In fact it had been carried out at a theoretical level already in Chapter 5 of [20] by means of the precursor of the Monotone Dialectica introduced in [16]. The treatment in [21] is even more platonic, by means of a good number of meta-theorems. We took the challenge to use a machine extraction in order to analyse on the computer a number of concrete instances of the HeExtEq example.

Definition 2.1 [[26], Section 2.7.2, adapted to the **T** presentation from [10]] The *extensional equality* at type $\sigma \equiv \sigma_1 \ldots \sigma_n \mathbb{N}$, denoted $=_\sigma$, is defined by

$$x =_{\mathbb{N}} y \;:\equiv\; \mathtt{at}(= xy)$$
$$x =_\sigma y \;:\equiv\; \forall z_1^{\sigma_1} \ldots z_n^{\sigma_n} (x z_1 \ldots z_n =_{\mathbb{N}} y z_1 \ldots z_n) \quad ,$$

where $=$ is defined in [10] as the usual equality boolean function on $\mathbb{N} \times \mathbb{N}$. It is immediate that $x =_{\rho\tau} y \equiv \forall z^\rho (xz =_\tau yz)$. As a parallel with the majorizability relation (see Definition 1.3), the *hereditarily extensional equality* is defined over the **T** type structure by

$$x \approx_{\mathbb{N}} y \;:\equiv\; x =_{\mathbb{N}} y$$
$$x \approx_{\rho\tau} y \;:\equiv\; \forall z_1^\rho, z_2^\rho (z_1 \approx_\rho z_2 \rightarrow x z_1 \approx_\tau y z_2) \quad ,$$

Definition 2.2 [Minimal Arithmetic] We denote by $\mathtt{WeZ_m}$ the system $\mathtt{WeZ}^\exists$ without the strong $\exists$ and also without the Ex-Falso-Quodlibet axiom $\bot \rightarrow F$, hence with an underlying Minimal Logic (in the sense of [15]) substructure.

[14] Via the boolean term associated (see Definition 1.4) to the *MD-radical* formula $A_{\mathtt{MD}}$ (a quantifier-free formula) which is at its turn associated to the formula A via Definition 1.5.

Proposition 2.3 ([20] - 5.13 or [21] - 8.17, adapted)
Let t^ρ be a closed term of Gödel's $\mathbf{T}$. Then $\mathtt{WeZ_m} \vdash t \approx_\rho t$.

Proof: By induction on the combinatorial structure of t, since closed terms of Gödel's $\mathbf{T}$ can be expressed[15] as built by application only (i.e., without lambda-abstraction) from 0, `Suc`, Gödel's recursor $\mathcal{R}$ and combinators Σ and Π. □

Corollary 2.4 ([20,21]) Let $t^{(\mathbb{N}\to\mathbb{N})\to(\mathbb{N}\to\mathbb{N})}$ be a closed $\mathbf{T}$-term. Since

$$\mathtt{WeZ_m} \;\vdash\; \forall x^{\mathbb{N}\to\mathbb{N}}, y^{\mathbb{N}\to\mathbb{N}}\, [\, x =_{\mathbb{N}\to\mathbb{N}} y \;\leftrightarrow\; x \approx_{\mathbb{N}\to\mathbb{N}} y\,]$$

it immediately follows that

$$\mathtt{WeZ_m} \;\vdash\; \forall x^{\mathbb{N}\to\mathbb{N}}, y^{\mathbb{N}\to\mathbb{N}}\, [\, x =_{\mathbb{N}\to\mathbb{N}} y \;\to\; t(x) =_{\mathbb{N}\to\mathbb{N}} t(y)\,] \quad .$$

Proposition 2.5 ([20] - 5.15 or [21] - 8.19, adapted) Let $t^{(\mathbb{N}\to\mathbb{N})\to(\mathbb{N}\to\mathbb{N})}$ be a closed term of Gödel's $\mathbf{T}$. Then t is uniformly continuous on every closed interval $B_y :\equiv \{x^{\mathbb{N}\to\mathbb{N}} \,|\, \forall z^{\mathbb{N}}.\; y(z) \succeq_{\mathbb{N}} x(z)\}$ with a modulus of uniform continuity which is effectively synthesizable (uniformly in $y^{\mathbb{N}\to\mathbb{N}}$) as a closed term $\widetilde{t}(y)^{\mathbb{N}\to\mathbb{N}}$ of $\mathbf{T}$, i.e., one can extract (by `MD`-interpretation) a closed $\mathbf{T}$-term $\widetilde{t}^{\,(\mathbb{N}\to\mathbb{N})\to(\mathbb{N}\to\mathbb{N})}$ s.t.:

$$\mathtt{WeZ_m} \;\vdash\; \forall y\, \forall x_1, x_2 \in B_y\, \forall k^{\mathbb{N}}\, [\, \bigwedge_{i=0}^{\widetilde{t}(y)(k)} x_1(i) =_{\mathbb{N}} x_2(i) \;\to\; \bigwedge_{j=0}^{k} t(x_1)(j) =_{\mathbb{N}} t(x_2)(j)\,]$$

Proof: Straightforward from Corollary 2.4 and Theorem 1.6, see [20,21] for details (in the Hilbert-style setting) of the proof originally introduced in [17]. □

The `HeExtEq` example was implemented in `MinLog` [9] in the sense that a minimal arithmetic `MinLog` proof of

$$\forall x^{\mathbb{N}\to\mathbb{N}}, y^{\mathbb{N}\to\mathbb{N}}\, [\, x =_{\mathbb{N}\to\mathbb{N}} y \;\to\; t(x) =_{\mathbb{N}\to\mathbb{N}} t(y)\,]$$

is mechanically generated for each particular $\mathbf{T}$-term $t^{(\mathbb{N}\to\mathbb{N})\to(\mathbb{N}\to\mathbb{N})}$ by a `Scheme` [23] procedure which takes as argument such a concrete `MinLog` $\mathbf{T}$-term t.

3 The *light* Monotone Dialectica *interpretation*

Our approach for the `MinLog` extraction of the generic modulus of uniform continuity $\widetilde{t}$, given the concrete `MinLog` term t is different from the letter of Proposition 2.5. It amounts in fact to the design of a new variant of the `MD`-interpretation, which combines those features of the pre-existing versions due to Kohlenbach[16] which turn out to be useful on the machine.

[15] Lemma 2.6 of [20] gives such a *syntactic* translation from λ-terms to combinatorial terms.

[16] We distinguish three such variants of the Monotone Dialectica interpretation, which were introduced in (chronologically ordered) [16], [18] and finally [19]. See also Zucker's chapter VI in [26], particularly its sections 8.3-6, for a raw, unformalized and quite primitive form of `MD`-interpretation.

We here name *light Monotone Dialectica* (abbreviated `LMD`-interpretation and even shorter, `LMDI`) this optimization of Kohlenbach's `MD`-interpretation for the extraction of majorants in `NbE`-normal form [17]. Hence the particularity of the new light `MD`-interpretation is the production of terms in normal form. In general, the normal form of a term may show to be (much) bigger than its more compact representation by means of lambda-abstractions. But on the other hand normalization may eliminate many redundant parts of the lambda-terms. Our practical experience with the automated, machine program-extraction, shows that the latter situation appears more often in our experiments, in particular for the `HeExtEq` case.

The key features of this novel form of `MD`-interpretation are the following:

(i) The terms extracted at each step of the recursion over the input proof structure are neither exact realizers, nor majorants, but *partial majorants*, in the sense that only the persistent contractions are treated like in [18].

(ii) An `NbE`-normalization (see [3,4,5] for the original `NbE`) of such extracted partial majorants is carried out for optimization purposes after the proof mining of the conclusion at each Implication Elimination (aka Modus Ponens) application. This recurrent form of partial normalization turns out to bring a huge improvement w.r.t. the one single final call-by-value `NbE` normalization process in situations of long sequences of nested Modus Ponens. We named this technique [18] "Normalization during Extraction" (abbreviated "`NdE`"), see [13] for a short account. The `HeExtEq` proof (described in Section 2 above) does actually contain quite long sequences of nested Modus Ponens.

(iii) The final such extracted partial majorant is `NbE`-normalized and then its majorant is built like in [16], but using the majorant for Gödel's recursor $\mathcal{R}$ from [19].

4 Machine results for the `HeExtEq` case-study in `MinLog`

We used our light Monotone Dialectica `MinLog` extraction modules which are available within the special [19] `MinLog` distribution [9]. We applied the `LMDI` extraction on the `MinLog` `HeExtEq` proof for the following concrete instances of the term t:

- The simple sum: $f, k \mapsto f(0) + \ldots + f(k)$.
- The double sum: $f, k \mapsto f(f(0)) + \ldots + f(f(k))$.
- The triple sum: $f, k \mapsto f(f(f(0))) + \ldots + f(f(f(k)))$.

In the case of the simple sum, the machine output is, as expected, the identity function, regardless of the actual f, hence the functional $f, k \mapsto k$. Also for the double sum, the outcome is the expected one, namely

$$f, k \;\mapsto\; max\{k\,,\, f(0)\,,\, \ldots\,,\, f(k)\} \quad .$$

[17] Here "`NbE`" is the usual acronym for "Normalization by Evaluation". See [3,4,5] for the original call-by-value `NbE` normalization technique.

[18] Which is a recurrent form of Partial Evaluation. See the volume [7] for accounts of the partial evaluation programming methodology.

[19] Our Dialectica modules are for the moment not compatible with the official `MinLog` distribution from [25].

On the contrary, for the triple sum, the mathematician needs to work a good number of minutes to produce the following *optimal* result

$$f, k \ \mapsto \ max\{k\,,\, f(0)\,,\, f(1)\,,\, \ldots\,,\, f(max\{k\,,\, f(0)\,,\, f(1)\,,\, \ldots\,,\, f(k)\})\} \qquad (1)$$

The machine produces in less than one minute an output which can be isomorphically adapted for display as follows:

$$f, k \ \mapsto \ max\{k\,,\, f(0)\,,\, \ldots\,,\, f(k)\,,\\ max\{f(0)\,,\, \ldots\,,\, f(max\{f(0)\,,\, \ldots\,,\, f(k)\})\}\} \qquad (2)$$

It is easy to notice that the machine-yielded expression (2) is immediately equivalent to the more human expression (1). Note also that in the context of a pointwise continuity demand, the optimal answer would be

$$f, g, k \ \mapsto \ max\{k\,,\, f(0)\,,\, f(1)\,,\, \ldots\,,\, f(k)\,,\, max\{f(f(0))\,,\, f(f(1))\,,\, \ldots\,,\, f(f(k))\}\}$$

which is strictly lower than the machine (or human) optimal output for the case of uniform continuity. In fact, while first trying to solve by brain the triple sum problem, we first erroneously thought that this were a modulus of uniform continuity, which is not the case. We later produced (1) by simplifying the machine outcome (2) and after some checks we realized the error. Hence we could produce a correct answer only with the help of the computer extraction.

Notwithstanding, right now a pattern can be noticed by the human in the solution of the `HeExtEq` problem for terms $t_l :\equiv \lambda f, k.\, f^{(l)}(0) + \ldots + f^{(l)}(k)$, with $f^{(l)}(i) :\equiv f(f \ldots (f(i)))$, where f appears l times on the right-hand side. We write again the above moduli of uniform continuity for t_l, with $l := 1, 2, 3$:

$$\begin{aligned} \widetilde{t_1}(f,k) \ &\equiv \ k \\ \widetilde{t_2}(f,k) \ &\equiv \ max\{k, f(0), \ldots, f(\widetilde{t_1}(f,k))\} \\ \widetilde{t_3}(f,k) \ &\equiv \ max\{k, f(0), \ldots, f(\widetilde{t_2}(f,k))\} \\ &\ldots\ldots\ldots\ldots\ldots\ldots \end{aligned}$$

We thus immediately infer the generic (recursive) solution for every $l \in \mathbb{N}$:

$$\widetilde{t_{l+1}}(f,k) \ \equiv \ max\{k, f(0), \ldots, f(\widetilde{t_l}(f,k))\}$$

The verification that $\widetilde{t_l}$ is the optimal modulus of uniform continuity for t_l is now an easy exercise, which we leave to the reader (see [11] for the solution).

5 Conclusions and future work

More such `MinLog` extractions of moduli of uniform continuity for other various concrete instances of the input term t can and ought to be performed. The light `MD`-interpretation should be mathematically formalized, in synthesis with the *light* optimization of Gödel's Dialectica from [10]. It might be that the latter improvement applies also in the case of the `HeExtEq` proof. This issue should be researched with high priority. Also a complete mathematical formulation of the Normalization during Extraction (NdE) ought to be given.

Acknowledgements: We would like to thank Prof. U. Kohlenbach for having suggested to us that the HeExtEq example may produce interesting results on the Computer. We also thank Prof. H. Schwichtenberg for having suggested to us that the already available formulations of the Monotone (or Bounded) Dialectica may not be satisfying enough from the computer-applied viewpoint.

References

[1] Avigad, J. and S. Feferman, *Gödel's functional ('Dialectica') interpretation*, in: S. Buss, editor, *Handbook of Proof Theory*, Studies in Logic and the Foundations of Mathematics **137**, Elsevier, 1998, pp. 337–405.

[2] Berger, U., W. Buchholz and H. Schwichtenberg, *Refined program extraction from classical proofs*, Annals of Pure and Applied Logic **114** (2002), pp. 3–25.

[3] Berger, U. and H. Schwichtenberg, *An inverse of the evaluation functional for typed λ–calculus*, in: R. Vemuri, editor, *Proceedings of the Sixth IEEE Symposium on Logic in Computer Science*, IEEE Computer Society Press, Los Alamitos, 1991, pp. 203–211.

[4] Berger, U., M. Eberl and H. Schwichtenberg, *Normalization by Evaluation*, in: B. Möller and J. Tucker, editors, *Prospects for Hardware Foundations*, LNCS **1546**, Springer Verlag, 1998, pp. 117–137.

[5] Berger, U., M. Eberl and H. Schwichtenberg, *Term rewriting for normalization by evaluation*, Information and Computation **183** (2003), pp. 19–42, International Workshop on Implicit Computational Complexity (ICC'99).

[6] Bezem, M., *Strongly majorizable functionals of finite type: A model for bar-recursion containing discontinuous functionals*, J. of Symb. Logic **50** (1985), pp. 652–660.

[7] Danvy, O., R. Glück and P. Thiemann, editors, "Partial Evaluation. Dagstuhl Castle, Germany, February 1996", LNCS **1110**, Springer Verlag, 1996.

[8] Gödel, K., *Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes*, Dialectica **12** (1958), pp. 280–287.

[9] Hernest, M.-D., *MinLog for Dialectica program-extraction*, Free software - code source @ http://www.brics.dk/~danher/MinLogForDialectica, For the official MinLog see [25].

[10] Hernest, M.-D., *Light Functional Interpretation*, LNCS **3634**, Springer Verlag, 2005, pp. 477 – 492, Computer Science Logic - CSL'2005.

[11] Hernest, M.-D., *Feasible programs from (non-constructive) proofs by the light (monotone) Dialectica interpretation*, PhD Thesis, École Polytechnique and University of Munich (2006), In preparation, draft available @ http://www.brics.dk/~danher/teza/.

[12] Hernest, M.-D., *Light Dialectica program extraction from a classical Fibonacci proof*, in: *Proceedings of DCM'06 at ICALP'06*, Electronic Notes in Theoretical Computer Science (ENTCS, 2007), 10pp., Accepted for publication, Downloadable @ http://www.brics.dk/~danher/.

[13] Hernest, M.-D., *NdE - Normalization during Extraction*, Regular Abstract, Local Proceedings of CiE'06 (Computability in Europe 2006), available in the author's web-page, see http://www.brics.dk/~danher/. Full paper in preparation.

[14] Howard, W., *Hereditarily majorizable functionals of finite type*, In [26], appendix chapter, pages 454–461.

[15] Johansson, I., *Der Minimalkalkül, ein reduzierter intuitionistischer Formalismus*, Compositio Matematica **4** (1936), pp. 119–136.

[16] Kohlenbach, U., *Effective bounds from ineffective proofs in analysis: an application of functional interpretation and majorization*, J. of Symb. Logic **57** (1992), pp. 1239–1273.

[17] Kohlenbach, U., *Pointwise hereditary majorization and some applications*, Archive for Mathematical Logic **31** (1992), pp. 227–241.

[18] Kohlenbach, U., *Analysing proofs in Analysis*, in: W. Hodges, M. Hyland, C. Steinhorn and J. Truss, editors, *Logic: from Foundations to Applications, Keele, 1993*, European Logic Colloquium (1996), pp. 225–260.

[19] Kohlenbach, U., *Mathematically strong subsystems of analysis with low rate of growth of provably recursive functionals*, Archive for Mathematical Logic **36** (1996), pp. 31–71.

[20] Kohlenbach, U., *Proof interpretations*, Technical report BRICS LS-98-1, DAIMI, Department of Computer Science, University of Aarhus, Aarhus, Denmark (1998), Free downloadble @ http://www.brics.dk/LS/Abs/BRICS-LS-Abs/BRICS-LS-Abs.html.

[21] Kohlenbach, U., *Proof Interpretations and the Computational Content of Proofs*, Latest version in the author's web page (April 2006), vii + 420pp.

[22] Kohlenbach, U. and P. Oliva, *Proof mining: a systematic way of analysing proofs in Mathematics*, Proc. of the Steklov Institute of Mathematics **242** (2003), pp. 136–164.

[23] Cadence Research Systems, *(Petite) Chez Scheme*, http://www.scheme.com (2006).

[24] Schwichtenberg, H., *Minimal logic for computable functions*, Lecture course on program-extraction from (classical) proofs. Available in the `MinLog` distribution [25].

[25] Schwichtenberg, H. and Others, *Proof- and program-extraction system `MinLog`*, Free software - code source and documentation @ http://www.minlog-system.de/.

[26] Troelstra, A., editor, "Metamathematical investigation of intuitionistic Arithmetic and Analysis", Lecture Notes in Mathematics (LNM) **344**, Springer-Verlag, 1973.