A way around Luckhardt's

elimination-of-extensionality procedure in the mining of proofs that use the non-standard analytical axiom **F**

Mircea-Dan Hernest

Comp Logic @ Universität Innsbruck

PCC'07 Talk in Swansea, 14 April 2007

Short summary of our work on Monotone Dialectica

1. New variant (optimization) of Kohlenbach's monotone Dialectica interpretation, named *"Light" Monotone Dialectica (LMD)*.
2. First presentation of the (Light) Monotone Dialectica in Natural Deduction style, as an elaborate transformation of Jørgensen's recent adaptation of the *pure* Gödel's Dialectica interpretation.
3. The *LMD* includes a full treatment of Berger's *ncm* quantifiers.
4. Some contractions are wiped out from the raw extracted terms. The (full) elimination of some Contractions still is very important also for Monotone Dialectica, hence not just for Pure Dialectica.
5. First-time implementation of the (Light) Monotone Dialectica on the computer, in a variant of MINLOG system. Theoretical results supported by a number of concrete examples treated on computer by means of the optimized (light) program-extraction technique.
6. A theory of synthesis of feasible, poly-time computable programs is developed. Two pre-existent frameworks due to Cook-Urquhart-Ferreira-Oliva and respectively Kohlenbach are crossbreeded into a *poly-time bounded Analysis*. No new practical applications here!

Outline of this Presentation

1. Introduction
   - Synthesis of verified programs by Proof Interpretations
   - Term system, Logic and Arithmetics for Program Extraction
2. From Gödel's Dialectica to Light (Monotone) Dialectica
   - The pure and the light Gödel's functional interpretation
   - The Contraction Problem –> Achilles' heel for any Extraction !
   - The Light Monotone Dialectica majorant extraction
3. The way around the elimination-of-extensionality procedure

Outline

Why $\forall x \exists y\, G(x, y)$ specifications ? [$G \equiv$ Goal Formula]

The generic program-extraction problem and its very basic generic terminology

- Specifications describe the wanted behavior of our Program.
- Programs have inputs – denoted x and outputs – denoted y .
- Therefore specifications are formulas $\forall x\, \exists y\, G(x, y)$ where
- $G(x, y)$ is a formula describing the desired relationship between the given input x and the desired output y . Moreover, there
- Exists a proof $\mathcal{P}$ of $\forall x\, \exists y\, G(x, y)$ in some logical system $\mathcal{S}$.
- We want to be able to *uniformly* produce by an Algorithm a program t which *realizes* the given specification, i.e., such that $\forall x\, G(x, t(x))$ is provable in some (other) logical system $\mathcal{S}'$.
- Such Algorithms taking inputs $\mathcal{P}$ are called "*Program Extraction procedures*" and the term t is called *extracted program*.
- Formula $G(x, y)$ may be *generally* arbitrary only when $\mathcal{S}$ is *constructive* and the **existential** quantification over y is **strong**

Outline

The term system – a lambda-variant of Gödel's **T**

1. All finite types generated from $\mathbb{N}$ and $\mathbb{B}$ by the rule $\sigma, \tau \mapsto (\sigma\tau)$
2. $tt^{\mathbb{B}}$ (boolean truth), $ff^{\mathbb{B}}$ (boolean falsity), $\mathbf{If}_{\mathbb{B}}$ (boolean binary test)
3. $And^{\mathbb{B}\mathbb{B}\mathbb{B}} :\equiv \lambda p, q.\, \mathbf{If}_{\mathbb{B}}\ p\ q\ ff$ $\quad$ $Imp^{\mathbb{B}\mathbb{B}\mathbb{B}} :\equiv \lambda p, q.\, \mathbf{If}_{\mathbb{B}}\ p\ q\ tt$
4. $0^{\mathbb{N}}$ (zero), $Suc^{\mathbb{N}\mathbb{N}}$ (successor) and Gödel's recursor $\mathbf{R}_{\tau}^{\tau(\mathbb{N}\tau\tau)\mathbb{N}\tau}$
5. $=^{\mathbb{N}\mathbb{N}\mathbb{B}}$ (equality), $\geq^{\mathbb{N}\mathbb{N}\mathbb{B}}$ (inequality), $Max^{\mathbb{N}\mathbb{N}\mathbb{N}}$ (maximum)
6. Combinators at all types are *defined* in terms of λ-abstraction:

$$\Sigma \ :\equiv\ \lambda x, y, z.\, x\, z\, (y\, z) \qquad \Pi \ :\equiv\ \lambda x, y.\, x$$

7. $at^{\mathbb{B}}$ is the *unique* predicate symbol of WeZ_m – one $\mathbb{B}$ argument
8. *Extensionally* defined equality and inequality (below $\sigma \in \{\mathbb{B}, \mathbb{N}\}$)

$$\begin{aligned}
s =_{\mathbb{N}} t &:\equiv at(= s\, t) & s =_{\mathbb{B}} t &:\equiv at(s) \leftrightarrow at(t)\\
s \geq_{\mathbb{N}} t &:\equiv at(\geq s\, t) & s \geq_{\mathbb{B}} t &:\equiv at(t) \rightarrow at(s)\\
s =_{\sigma_1 \ldots \sigma_n \rightarrow \sigma} t &:\equiv \forall x_1^{\sigma_1} \ldots x_n^{\sigma_n} (s\, x_1 \ldots x_n =_{\sigma} t\, x_1 \ldots x_n)\\
s \geq_{\sigma_1 \ldots \sigma_n \rightarrow \sigma} t &:\equiv \forall x_1^{\sigma_1} \ldots x_n^{\sigma_n} (s\, x_1 \ldots x_n \geq_{\sigma} t\, x_1 \ldots x_n)
\end{aligned}$$

Majorizability and Hereditarily Extensional Equality

$$x\ maj_{\mathbb{N}}\ y \quad :\equiv \quad x \geq_{\mathbb{N}} y \quad :\equiv \quad at(\geq x^{\mathbb{N}}\ y^{\mathbb{N}})$$

$$x \geq_{\sigma\tau} y \quad \equiv \quad \forall z^{\sigma}\,(xz \geq_{\tau} yz)$$

$$x\ maj_{\sigma\tau}\ y \quad :\equiv \quad \forall z_1^{\sigma}, z_2^{\sigma}\,(z_1\ maj_{\sigma}\ z_2 \ \rightarrow\ xz_1\ maj_{\tau}\ yz_2)$$

$0\ maj_{\mathbb{N}}\ 0$, $Suc\ maj_{\mathbb{N}\mathbb{N}}\ Suc$, $\boxed{\Sigma\ maj\ \Sigma\ ,\ \Pi\ maj\ \Pi\ \text{and}\ \mathbf{R}^{M}\ maj\ \mathbf{R}}$

$$WeZ_m \ \vdash\ t^{\star}\ maj_{\sigma\tau}\ t \ \wedge\ s^{\star}\ maj_{\sigma}\ s \ \Longrightarrow\ t^{\star}s^{\star}\ maj_{\tau}\ ts$$

$$x \approx_{\mathbb{N}} y \quad :\equiv \quad x =_{\mathbb{N}} y \quad :\equiv \quad at(= x^{\mathbb{N}}\ y^{\mathbb{N}})$$

$$x =_{\sigma\tau} y \quad \equiv \quad \forall z^{\sigma}\,(xz =_{\tau} yz)$$

$$x \approx_{\sigma\tau} y \quad :\equiv \quad \forall z_1^{\sigma}, z_2^{\sigma}\,(z_1 \approx_{\sigma} z_2 \ \rightarrow\ xz_1 \approx_{\tau} yz_2)$$

$0 \approx_{\mathbb{N}} 0$, $Suc \approx_{\mathbb{N}\mathbb{N}} Suc$, $\boxed{\Sigma \approx \Sigma\ ,\ \Pi \approx \Pi\ \text{and}\ \mathbf{R} \approx \mathbf{R}}$

$$WeZ_m \ \vdash\ t^{\star} \approx_{\sigma\tau} t \ \wedge\ s^{\star} \approx_{\sigma} s \ \Longrightarrow\ t^{\star}s^{\star} \approx_{\tau} ts$$

The Light (Mon.) Dialectica interpretation of formulas

$$
\begin{aligned}
A^{\mathbf{D}} &:\equiv (A_{\mathbf{D}} :\equiv A) \text{ for prime formulas } A \\
(A \wedge B)^{\mathbf{D}} &:\equiv \exists \underline{x}, \underline{u} \forall \underline{y}, \underline{v} \, [(A \wedge B)_{\mathbf{D}} :\equiv A_{\mathbf{D}}(\underline{x}; \underline{y}; \underline{a}) \wedge B_{\mathbf{D}}(\underline{u}; \underline{v}; \underline{b})] \\
(A \to B)^{\mathbf{D}} &:\equiv \exists \underline{Y}, \underline{U} \forall \underline{x}, \underline{v} \, [(A \to B)_{\mathbf{D}} :\equiv \\
&\qquad A_{\mathbf{D}}(\underline{x}; \underline{Y}(\underline{x}, \underline{v})) \to B_{\mathbf{D}}(\underline{U}(\underline{x}); \underline{v})] \\
(\exists z A(z, \underline{a}))^{\mathbf{D}} &:\equiv \exists z^{\dagger}, \underline{x} \forall \underline{y} \, [(\exists z A(z, \underline{a}))_{\mathbf{D}}(z^{\dagger}, \underline{x}; \underline{y}; \underline{a}) :\equiv A_{\mathbf{D}}(\underline{x}; \underline{y}; z^{\dagger}, \underline{a})] \\
(\overline{\exists} z A(z, \underline{a}))^{\mathbf{D}} &:\equiv \exists \underline{x} \forall \underline{y} \, [(\overline{\exists} z A(z, \underline{a}))_{\mathbf{D}}(\underline{x}; \underline{y}; \underline{a}) :\equiv \exists z \, A_{\mathbf{D}}(\underline{x}; \underline{y}; z, \underline{a})] \\
(\forall z A(z, \underline{a}))^{\mathbf{D}} &:\equiv \exists \underline{X} \forall z^{\dagger}, \underline{y} \, [(\forall z A(z, \underline{a}))_{\mathbf{D}}(\underline{X}; z^{\dagger}, \underline{y}; \underline{a}) :\equiv A_{\mathbf{D}}(\underline{X}(z^{\dagger}); \underline{y}; z^{\dagger}, \underline{a})] \\
(\overline{\forall} z A(z, \underline{a}))^{\mathbf{D}} &:\equiv \exists \underline{x} \forall \underline{y} \, [(\overline{\forall} z A(z, \underline{a}))_{\mathbf{D}}(\underline{x}; \underline{y}; \underline{a}) :\equiv \forall z \, A_{\mathbf{D}}(\underline{x}; \underline{y}; z, \underline{a})]
\end{aligned}
$$

Here $\cdot \mapsto \cdot^{\dagger}$ is a mapping which assigns to every given variable z a completely new variable $z^{\dagger}$ which has the same type of z.

Quantifiers $\overline{\forall}$ and $\overline{\exists}$ are called *ncm* ("non-computational-meaning")

System WeZ_m. Implication Introduction with Contraction

1. WeZ_m - Weakly extensional Minimal Arithmetic with $\geq$ and *Max*
2. Minimal Arithm. $\equiv$ Heyting Arithm. in all finite types $HA^{\omega} \setminus \bot \rightarrow F$
3. WeZ_m - underlying Logic is in Natural Deduction, not Hilbert-style!
4. $\dfrac{[u : A] \ldots / B}{A \rightarrow B} \rightarrow^{+}$ Particular set of instances of A in the same parcel (assumption variable) u get discharged. If at least two A's get discharged then one has a *logical Contraction*. If moreover A contains at least one positive universal or a negative existential quantifier then one has a *"computationally relevant" Contraction*.
5. This *computational relevance* is meant more relative to Gödel's Dialectica and in a lesser measure relative to Monotone Dialectica

$$\{A_{\mathbf{D}}(z; T_i(\underline{z}, \underline{x}, y))\}_{i=1}^{n+1}, \{C^i_{\mathbf{D}}(x_i; T_i(\underline{z}, \underline{x}, y))\}_{i=n+2}^{m} \vdash B_{\mathbf{D}}(T(\underline{z}, \underline{x}); y)$$

Same tuple z produced by $2 \leq n+1 \leq m$ discharged instances of A

If $\{T_i\}_{i=1}^{n+1}$ non-null (A is Dialectica-relevant) $\Rightarrow$ *Equalization* is a must!

Extensionality/Compatibility and Induction rules

$E_{\sigma,\tau} : \forall z^{\sigma\tau}, x^{\sigma}, y^{\sigma}.\ x =_{\sigma} y \rightarrow zx =_{\tau} zy$ – must be forbidden

$$\frac{\begin{array}{c} A_0 \\ \vdots \\ s =_{\sigma} t \end{array}}{B(s) \rightarrow B(t)}$$

$COMPAT_{\sigma}$ – with the restriction that all undischarged assumptions used in the proof of $s =_{\sigma} t$ (here denoted A_0) are quantifier-free

$$\frac{\begin{array}{cc} \emptyset & \emptyset \\ \vdots & \vdots \\ A(0) & \forall z\,(A(z) \rightarrow A(Sucz)) \end{array}}{\forall z\, A(z)}$$

IR_0 – equivalent to IA, IR in WeZ_m

$A(tt) \wedge A(ff) \rightarrow \forall p^{\mathbb{B}} A(p)$
(Boolean Induction Axiom)

$$\left.\begin{array}{l} \mathbf{R}_{\tau}\, x\, y\, 0 =_{\tau} x \\ \mathbf{R}_{\tau}\, x\, y\, (Suc\, z) =_{\tau} y(z, \mathbf{R}_{\tau}\, x\, y\, z) \end{array}\right\} : \mathtt{Ax}\mathbf{R}_{\tau}$$

Outline

Gödel's functional "Dialectica" interpretation

1. A translation of proofs which includes a translation of formulas.
2. $A(\underline{a}) \mapsto A^{\mathbf{D}} \equiv \exists \underline{x} \forall \underline{y}\, A_{\mathbf{D}}(\underline{x}; \underline{y}; \underline{a})$ with $\underline{a}$ all free vars of formula A
3. $A_{\mathbf{D}}$ is quantifier-free for Gödel's Dialectica, since decidability needed –> this no longer for Monotone setup $\Rightarrow$ *Bounded* Dialectica
4. Recursive syntactic translation from proofs in *Constructive Arithmetic* (or *Classical Arithmetic*, modulo the double-negation translation) to proofs in *Intuitionistic Arithmetic* such that positive occurrences of $\exists$ and negative occurrences of $\forall$ in the proof's conclusion get actually realized by terms in Gödel's **T**.
5. *Contraction Problem:* –> choose between a number of realizers according to a boolean term associated to the contraction formula; *Diller-Nahm:* –> postpone all choices to the very end by collecting all candidates and making a single final global choice; *Monotone Dialectica:* –> use a simple common upper bound (maximum majorant) of the candidates $\Longrightarrow$ extract *majorants*

Clear-cut practical example - classical Fibonacci

The semi-classical Fibonacci proof in MINLOG is a Natural Deduction proof of $\forall n\, \exists^{cl} k\; G(n,k) \;\equiv\; \forall n\, [(\forall k.\, G(n,k) \to \bot) \to \bot]$ from assumptions expressing that G is the graph of the Fibonacci function:

$$G(0,0) \& G(1,1) \& \mathit{Step} : \forall n,k,l.\, [G(n,k) \wedge G(n+1,l)] \to G(n+2,k+l)$$

$$[G,n_1]\pi_1\pi_2(\mathbf{R}_{\mathbb{N}=>\mathbb{N}@(\mathbb{N}@\mathbb{N})@(\mathbb{N}@\mathbb{N})}((0@0@0)@0@1)\{[n_2,p][\mathbf{If}[\mathbf{If}(G\pi_1\pi_1(p)\pi_1\pi_2\pi_1(p))$$
$$[\mathbf{If}(G(Suc\pi_1\pi_1(p))\pi_2\pi_2\pi_1(p))(G(Suc(Suc\pi_1\pi_1(p)))(\pi_1\pi_2\pi_1(p)+\pi_2\pi_2\pi_1(p)))tt]tt]$$
$$(n_2@\pi_2(p))(\pi_1(p))]@\pi_2\pi_2(p)@\pi_1\pi_2(p)+\pi_2\pi_2(p)\}n_1) \quad \underline{\textit{Goedels Dialectica Program}}$$

If-tests are only due to the Contraction formula, integrated in the extracted program

Contraction is here due to the Dialectica interpretation of the general Induction Rule *IR* in terms of the simpler Induction Rule IR_0. The contraction formula is just Step! This is because Step is an open assumption of the step case of *IR*. But we can use $\overline{\forall} n,k,l.$ hence get

MINLOG program extracted by Light Dialectica (after normalization)

$$[n_0]\pi_1(\mathbf{R}_{\mathbb{N}\to(\mathbb{N}@\mathbb{N})}(0@1)\{[n_1,p^{\mathbb{N}@\mathbb{N}}]\pi_2(p)@(\pi_1(p)+\pi_2(p))\}n_0)$$

Exactly the usual algorithm computing the n-th Fibonacci number.

Exact realizer synthesis by Dialectica Interpretations

Extraction and Soundness Theorem: There exists an algorithm which, given at input a $WeZ^{\exists,nc+}$ proof $\mathcal{P} : \{C^i\}_{i=1}^n \vdash A$ [hence of the conclusion formula A, from the *undischarged* assumption formulas $\{C^i\}_{i=1}^n$] will produce at output 1) the tuples of terms T and $\{T_i\}_{i=1}^n$ 2) the tuples of variables $\{x_i\}_{i=1}^n$ and y 3) the verifying proof

$$\mathcal{P}_{\mathbf{D}} : \{C^i_{\mathbf{D}}(x_i; T_i(\underline{x}, y))\}_{i=1}^n \vdash A_{\mathbf{D}}(T(\underline{x}); y)$$

in $WeZ^{\exists}$ – where $\underline{x} :\equiv x_1, \ldots, x_n$. Moreover,

1. variables $\underline{x}$ and y are all completely new (they do not occur in $\mathcal{P}$)
2. the free variables of T and $\{T_i\}_{i=1}^n$ are among the free variables of A and $\{C^i\}_{i=1}^n$ [this we named as "the *free variable condition* (**FVC**) for programs extracted by the Dialectica Interpretation"]

[$\Rightarrow$ $\underline{x}$, y do not occur free in the *extracted* terms $\{T_i\}_{i=1}^n$ and T]

Notice that: Terms T and $\{T_i\}_{i=1}^n$ are not necessarily closed !!!

Outline

Problem –> Implication Introduction with Contraction

$$\boxed{\frac{[u:\ A]\ \ldots\ /B}{A \to B}\ \to^{+}}$$

$n \geq 1,\ \ \underline{z} \equiv \overbrace{z, \ldots, z}^{n+1}$ and $\underline{x} \equiv x_{n+2}, \ldots, x_m$:

$$\{A_{\mathbf{D}}(z; T_i(\underline{z}, \underline{x}, y))\}_{i=1}^{n+1}\ ,\ \{C^i_{\mathbf{D}}(x_i; T_i(\underline{z}, \underline{x}, y))\}_{i=n+2}^{m} \vdash\ B_{\mathbf{D}}(T(\underline{z}, \underline{x}); y)$$

1) Same tuple z produced by $n+1 \leq m$ discharged instances of A

2) Case: tuples $\{T_i\}_{i=1}^{n+1}$ are non-null! Recall that $A_{\mathbf{D}}$ is quantifier-free

3) Since $\{T_i\}_{i=1}^{n+1}$ non-null $\implies$ their *equalization* is a *must*:

$$\mathbf{S}\ :\equiv\ \lambda \underline{x}, z, y\,.\,\mathbf{If}^n_\tau(\mathrm{t}^{\mathbf{D}}_A[z; T^1], \ldots, \mathrm{t}^{\mathbf{D}}_A[z; T^n], T_{n+1}(\underline{z}, \underline{x}, y), T^n, \ldots, T^1)$$

one can now cancel all $\{A_{\mathbf{D}}\}_{i=1}^{n+1}$ *by a single* $\to^{+}$ *in the verifying proof*

$$\{A_{\mathbf{D}}(z; \mathbf{S}(\underline{x}, z, y))\}_{i=1}^{n+1}\ ,\ \{C^i_{\mathbf{D}}(x_i; S_i(\underline{x}, z, y))\}_{i=n+2}^{m} \vdash\ B_{\mathbf{D}}(S(\underline{x}, z); y)$$

$$\{C^i_{\mathbf{D}}(x_i; S_i(\underline{x}, z, y))\}_{i=n+2}^{m} \vdash\ A_{\mathbf{D}}(z; \mathbf{S}(\underline{x}, z, y)) \to B_{\mathbf{D}}(S(\underline{x}, z); y)$$

Light Dialectica Interpretation of the rule of ncm-forall ($\overline{\forall}$) introduction

$$\frac{\mathcal{P} : A(z)}{\overline{\forall} z\, A(z)}\ \overline{\forall}_z^+$$

We are given that $\{C^i_{\mathbf{D}}(x_i; T_i(\underline{x}, y))\}^n_{i=1} \vdash A_{\mathbf{D}}(T(\underline{x}); y; z)$ where z is neither in $\cup^n_{i=1} \mathcal{V}_{\mathrm{f}}(C^i)$ (because of $VC_1(z)$) nor among $\underline{x}, y$ (due to the rules for variable naming in the **LD**-interpretation of a formula). The fact that z is not free in any of $T, \{T_i\}^n_{i=1}$ is ensured mainly by $VC_3(z, \mathcal{P})$ - the exception in $VC_3(z, \mathcal{P})$ is taken care of by the **FVC** applied to all sub-proofs of $\mathcal{P}$. Thus, even in the exceptional cases, z still is not free in any of $T, \{T_i\}^n_{i=1}$, because of the **FVC**. Since the pre-condition $VC_1(z)$ is established, we can apply a $\forall^+_z$ in the verifying proof to obtain, with the **FVC** thus clearly satisfied that $\{C^i_{\mathbf{D}}(x_i; T_i(\underline{x}, y))\}^n_{i=1} \vdash \forall z\, A_{\mathbf{D}}(T(\underline{x}); y; z)$.

- $VC_1(z)$: the variable z does not occur free in any of the undischarged assumptions of the proof of the premise of the rule;
- $VC_3(z, \mathcal{P})$: the variable z does not occur free in the instantiating terms t involved by the ForAll eliminations $\forall^-_{\bullet, t}$ from the proof $\mathcal{P}$ (so far Berger's restriction) *and* z is also not free in the computationally LD-relevant contraction formulas involved by Implication Introductions $\rightarrow^+$ from $\mathcal{P}$ *except* for the cases when ...

Outline

The Light Monotone Dialectica program extraction

Theorem: Majorant realizer synthesis by Light Monotone Dialectica

There exists an algorithm which, given at input a $WeZ_m^{\exists,nc+}$ 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 $\{C^i\}_{i=1}^n$] it will produce at output the following ($\underline{a} :\equiv a_1, \ldots, a_n, a'$):

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

$$\exists Y_1, \ldots Y_n, X\, [\, \bigwedge_{i=1}^n (\lambda \underline{a}.\, T_i^\star[\underline{a}])\ maj\ Y_i \ \wedge\ (\lambda \underline{a}.\, T^\star[\underline{a}])\ maj\ X \ \wedge$$
$$\forall \underline{a}, \underline{x}, y\, (\, \{\bigwedge_{i=1}^n C^i_{\mathbf{D}}(x_i; Y_i(\underline{a}, \underline{x}, y); a_i)\} \ \rightarrow\ A_{\mathbf{D}}(X(\underline{a}, \underline{x}); y; a')\,)\,]$$

Variables $\underline{x}$ and y do not occur in $\mathcal{P}$ (they are all completely new)

$\Longrightarrow$ $\underline{x}$ and y do not occur free in the *extracted* terms $\{T_i\}_{i=1}^n$ and T.

Let $\Delta, \Delta' \equiv \{ \forall x^\rho \exists y \leq_\sigma rx \forall z^\tau \ B^{nc}(x,y,z)\}$ be two distinct sets of sentences of this particular shape, where B^{nc} is a purely-*ncm* formula, i.e., it may contain only *ncm* quantifiers.

Moreover, the elements of Δ are restricted to formulas in which all positively (*ncm*-)universal and negatively (*ncm*-)existential quantified variables have type degree at most 2 and also all positively (*ncm*-)existential and negatively (*ncm*-)universal quantified variables have type degree at most 1.

Hence in particular, the regularly universal quantified variables x^ρ and z^τ have the restriction $dg(\rho), dg(\tau) \leq 2$, and also the regularly existential quantified variable y^σ has the restriction $dg(\sigma) \leq 1$.

Let $\mathcal{S}^\omega$ denote as usual the full *ZFC* set-theoretic type structure.

We further assume that $\mathcal{S}^\omega \models \Delta_{reg}$, where $\Delta_{reg} \equiv \{ \forall x^\rho \exists y \leq_\sigma rx \forall z^\tau \ B(x,y,z)\}$ is the direct regular-quantifier translation of Δ (here B is the usual full regular-quantifier translation of B^{nc}, obtained by replacing $\overline{\forall}$ with $\forall$ and $\overline{\exists}$ with $\exists$).

Let also $\mathcal{M}^\omega$ denote Bezem's type structure of all strongly majorizable functionals, as usual.

The formulas of Δ' are restricted only by $\mathcal{M}^\omega \models \Delta'_{reg}$. Let $WeZ_m^{\exists}$, $WeZ_m^{\exists,nc}$, $WeZ_m^{\exists,nc,c+}$ be the classical arithmetics for light monotone Dialectica from Chapter 1 of [1] and also let PbZ, PbZ^{c+} be the polynomial classical arithmetics from Chapter 3 of [1].

We further assume that $WeZ_m^{\exists,nc,c+}$ and PbZ^{c+} no longer contain any implicit Δ-kind of axiom set and for simplicity also not any kind of Π axiom set.

We leave as an easy exercise to the reader that one can add also an axiom set $\Pi \equiv \{\forall \underline{b} B^{nc}(\underline{b}) \mid \mathcal{S}^\omega \models \forall \underline{b} B(\underline{b})\}$ to which the same type restrictions as for Δ apply, see Section 3.2 of [1] for details.

If instead of a syntactic verifying proof, a simple guarantee that the verification holds in the full set-theoretic type structure $\mathcal{S}^\omega$ suffices, then the following extraction theorems can be established in the spirit of Theorem 4.9 of [2].

Theorem: Let $A_1(x^{\mathbb{N}\mathbb{N}}, k^{\mathbb{N}}, y^{\delta}, z^{\gamma})$ be a quasi-purely-existential formula of $WeZ_m^{\exists,nc}$ with x, k, y, z all its free variables, i.e., $A_1 \equiv \exists v A^{nc}(x^{\mathbb{N}\mathbb{N}}, k^{\mathbb{N}}, y^{\delta}, z^{\gamma}, v^{\alpha})$, and moreover such that $dg(\delta) \leq 1, dg(\gamma), dg(\alpha) \leq 2$ and further all positively *ncm*-universal and negatively *ncm*-existential quantified variables of A^{nc} have type degree at most 1 and also all positively *ncm*-existential and negatively *ncm*-universal quantified variables of A^{nc} have type degree at most 2. Let $s^{(\mathbb{N}\mathbb{N})\mathbb{N}\delta}$ be a closed term of $WeZ_m^{\exists}$. Let Δ and Δ' be the explicit sets of axiom sentences defined above (recall that $\mathcal{S}^{\omega} \models \Delta_{reg}$ and $\mathcal{M}^{\omega} \models \Delta'_{reg}$). Then there exists an (light monotone Dialectica) algorithm which from a given proof

$$WeZ_m^{\exists,nc,c+} + \Delta + \Delta' \quad \vdash \quad \forall x^{\mathbb{N}\mathbb{N}}\, \forall k^{\mathbb{N}}\, \forall y \leq_{\delta} sxk\, \exists z^{\gamma}\, A_1(x,k,y,z) \tag{1}$$

produces the closed term $\mathbf{t}^{(\mathbb{N}\mathbb{N})\mathbb{N}\gamma}$ of $WeZ_m^{\exists}$ such that

$$\mathcal{S}^{\omega} \models \forall x^{\mathbb{N}\mathbb{N}}\, \forall k^{\mathbb{N}}\, \forall y \leq_{\delta} sxk\, \exists z \leq_{\gamma} \mathbf{t}xk\, \widetilde{A_1}(x,k,y,z) \tag{2}$$

where $\widetilde{A_1}(x,k,y,z) \equiv \exists v A(x,k,y,z,v)$ is the regular-quantifier translation of A_1.

It is easy to check that the non-standard (i.e., not valid in $\mathcal{S}^\omega$) analytical axiom

$$\mathbf{F}^- \quad :\equiv \quad \forall \Phi^{\mathbb{N}(\mathbb{N}\mathbb{N})\mathbb{N}} \forall x^{\mathbb{N}\mathbb{N}\mathbb{N}\mathbb{N}} \exists y \leq_{\mathbb{N}\mathbb{N}\mathbb{N}\mathbb{N}} x \forall k^{\mathbb{N}} \forall z^{\mathbb{N}\mathbb{N}} \forall n^{\mathbb{N}}$$
$$[\wedge_{i<_{\mathbb{N}} n}(z\,i \leq_{\mathbb{N}} x\,k\,i) \rightarrow \Phi\,k\,(\lambda k^{\mathbb{N}}.\,\mathbf{If}_{\mathbb{N}}(k <_{\mathbb{N}} n)(z\,k)\circ) \leq_{\mathbb{N}} \Phi\,k\,(y\,k)]$$

can be included into the axiom set Δ', since $\mathbf{F}^-$ has the right Δ-shape and moreover $\mathcal{M}^\omega \models \mathbf{F}^-$ (see Remark 4.17 of [2]). On the other hand, although valid in $\mathcal{M}^\omega$ (see Proposition 4.6 of [2]), the stronger axiom

$$\mathbf{F} \;:\equiv\; \forall \Phi^{\mathbb{N}(\mathbb{N}\mathbb{N})\mathbb{N}} \forall x^{\mathbb{N}\mathbb{N}\mathbb{N}\mathbb{N}} \exists y \leq_{\mathbb{N}\mathbb{N}\mathbb{N}\mathbb{N}} x \forall k^{\mathbb{N}} \forall z \leq_{\mathbb{N}\mathbb{N}} yk\,(\Phi kz \leq_{\mathbb{N}} \Phi k(yk))$$

is not directly of Δ shape, because of the type-$\mathbb{N}$ negative universal quantifier expanded from the extensional definition of $z \leq_{\mathbb{N}\mathbb{N}} yk$. Nevertheless, $\mathbf{F}$ can be made into a Δ' axiom by using an *ncm*-universal quantifier instead of the regular universal quantifier which causes the trouble. Let

$$\mathbf{F^{nc}} \;:\equiv\; \forall \Phi^{\mathbb{N}(\mathbb{N}\mathbb{N})\mathbb{N}} \forall x^{\mathbb{N}\mathbb{N}\mathbb{N}\mathbb{N}} \exists y \leq_{\mathbb{N}\mathbb{N}\mathbb{N}\mathbb{N}} x \forall k^{\mathbb{N}} \forall z^{\mathbb{N}\mathbb{N}}$$
$$[\overline{\forall} l^{\mathbb{N}}(zl \leq_{\mathbb{N}} ykl) \rightarrow \Phi kz \leq_{\mathbb{N}} \Phi k(yk)]$$

be such an *ncm*-variant of $\mathbf{F}$, which is easily seen to be a Δ-shape axiom.

Since moreover $\mathcal{M}^\omega \models \mathbf{F}$, which is the direct regular-quantifier translation of $\mathbf{F^{nc}}$, i.e., $\mathbf{F} \equiv (\mathbf{F^{nc}})_{\mathbf{reg}}$, it follows that $\mathbf{F^{nc}}$ is also a Δ'-axiom.

Note that none of $\mathbf{F}^-$ and $\mathbf{F^{nc}}$ is an explicit Δ axiom because $\mathcal{S}^\omega \not\models \mathbf{F}^-$ and also $\mathcal{S}^\omega \not\models \mathbf{F} \equiv (\mathbf{F^{nc}})_{\mathbf{reg}}$
(see [2] for indications to the proofs of these).

One thus obtains, without using the elimination-of-extensionality procedure (in contrast to Theorem 4.9 of [2] which does use it), the following:

Corollary: Full admissibility of $\mathbf{F}^-$ and $\mathbf{F^{nc}}$ to the direct *LMD*-extraction
The previously-displayed **Theorem** holds as well in the variant when its hypothesis (1) is replaced by

$$WeZ_m^{\exists,nc,c+} + \Delta + \mathbf{F}^- + \mathbf{F^{nc}} \vdash \forall x^{\mathbb{N}\mathbb{N}} \forall k^{\mathbb{N}} \forall y \leq_\delta sxk \exists z^\gamma \; A_1(x,k,y,z)$$

Corollary: [Immediate adaptation to the Polynomial-Feasible case]

The previously-displayed Theorem and Corollary immediately adapt to the extraction of polynomial bounds in the sense of [2] in the following way. Assume that A_1 is a *PbZ* formula and that $s^{(\mathbb{N}\mathbb{N})\mathbb{N}\delta}$ is a closed term of *PbZ*. Then there exists an algorithm which from a given proof $PbZ^{c+} + \Delta + \Delta' \vdash \forall x^{\mathbb{N}\mathbb{N}} \forall k^{\mathbb{N}} \forall y \leq_\delta sxk \exists z^\gamma A_1(x,k,y,z)$ produces the syntactic polynomial $\overline{p}[x^{\mathbb{N}\mathbb{N}}, k^{\mathbb{N}}, u^{\mathbb{N}\mathbb{N}}, l^{\mathbb{N}}]^{\mathbb{N}} \in PbZ$ such that

$$\mathcal{S}^\omega \models \forall x^{\mathbb{N}\mathbb{N}} \forall k^{\mathbb{N}} \forall y \leq_\delta sxk \exists z \leq_\gamma \lambda u^{\mathbb{N}\mathbb{N}}, l^{\mathbb{N}}. \overline{p}[x^M, k, u^M, l]\ \widetilde{A_1}(x,k,y,z)$$

where u and l are (possibly empty) tuples determined by γ. Hence if $\gamma \equiv \mathbb{N}$ then p is a polynomial bound for z in x^M and k which is uniform w.r.t. y. All the above hold in particular for $\Delta' \equiv \{\mathbf{F}^-, \mathbf{F}^{\mathbf{nc}}\}$.

See Section 3.2.2 of [1] for terminology and more details . . . :)

Short complete Summary of the Results of my Thesis

1. New variant - optimization of Gödel's Dialectica interpretation
2. New presentation in Natural Deduction style, as an improvement of Jørgensen's recent adaptation of pure Gödel's Dialectica.
3. *Dialectica Light* includes treatment of Berger's *ncm* quantifiers
4. Some contractions are wiped out from the raw extracted terms
5. This "*light*" variant combines even more smoothly with Kohlenbach's "monotone" optimization of Dialectica. Full elimination of some Contractions still very important also for Monotone Dialectica
6. First-time formulation of Monotone Dialectica in Natural Deduction
7. Theoretical results supported by a number of concrete examples treated on the computer by means of the novel techniques. First-time implementation of a Monotone-Dialectica on the computer
8. Theoretical&Practical comparison with BBS refined A-translation
9. A theory of synthesis of feasible, poly-time computable programs is developed. Two pre-existent frameworks due to Cook-Urquhart-Ferreira and respectively Kohlenbach are crossbreeded into a "poly-time bounded Analysis". No new practical applications here!

Short List of related Papers I


M.-D. Hernest.
Feasible programs from (non-constructive) proofs by the light (monotone) Dialectica interpretation.
PhD Thesis, École Polytechnique, December 2006.
FirstRevised version available @ http://www.brics.dk/~danher/teza/


U. Kohlenbach.
Mathematically strong subsystems of analysis with low rate of growth of provably recursive functionals.
Archive for Mathematical Logic, 36:31–71, 1996.


U. Kohlenbach.
Pointwise hereditary majorization and some applications.
Archive for Mathematical Logic, 31:227–241, 1992.


U. Kohlenbach.
Proof Interpretations and the Computational Content of Proofs.
Lecture Course, latest version in the author's web page.

Short List of related Papers II


U. Kohlenbach and P. Oliva.
Proof Mining: a systematic way of analysing proofs in Mathematics.
Proc. of the Steklov Inst. of Mathem., 242:136–164, 2003.


U. Kohlenbach.
Analysing proofs in Analysis.
In *Logic: from Foundations to Applications, Keele, 1993*, European Logic Colloquium, pages 225–260. Oxford University Press, 1996.


M.-D. Hernest.
Light Dialectica program extraction from a classical Fibonacci proof Proceedings of DCM@ICALP'06, ENTCS (2007), 10pp.


M.-D. Hernest. Light Functional Interpretation.
CSL 2005 - In *LNCS* 3634 pp. 477 – 492, July 2005.