Proof mining: Lecture 3

Applications of proof theory in mathematics

Thomas Powell

Technische Universität Darmstadt

Autumn School on Proof and Computation

Fischbachau

21 September 2018

Outline

The Gödel-Gentzen negative translation

Let A be a formula in predicate logic. We define the negative translation of A by

$$A^N :\equiv \neg\neg A^*$$

where A^* is defined inductively as

$$\begin{aligned} A^* &:\equiv A \text{ if } A \text{ is a prime formula} \\ (A \Box B)^* &:\equiv A^* \Box B^* \text{ if } \Box \in \{\wedge, \vee, \rightarrow\} \\ (\exists x A)^* &:\equiv \exists x A^* \\ (\forall x A)^* &\equiv \forall x \neg\neg A^* \end{aligned}$$

Soundness of the negative translation

The negative translation obeys the following general pattern: Suppose that

$$\mathcal{P}_{\text{class}} \vdash A$$

for some classical theory $\mathcal{P}_{\text{class}}$. Then

$$\mathcal{P} \vdash A^N$$

where $\mathcal{P}$ is the intuitionistic version of that theory.

In particular, this is true for Peano/Heyting arithmetic.

Theorem
If $\mathrm{PA}^\omega \vdash A$ *then* $\mathrm{HA}^\omega \vdash A^N$.

Proof.
Induction over the structure of derivations in PA^ω. □

The negative translation of ∀∃∀ formulas

Suppose that $A :\equiv \forall k \exists n \forall m P(k,n,m)$ for $P(k,n,m)$ quantifier-free. Then

$$\begin{aligned} A^N &\equiv \neg\neg A^* \\ &\equiv \neg\neg(\forall k \exists n \forall m P(k,n,m))^* \\ &\equiv \neg\neg\forall k \neg\neg(\exists n \forall m P(k,n,m))^* \\ &\equiv \neg\neg\forall k \neg\neg\exists n \forall m \neg\neg P(k,n,m). \end{aligned}$$

This looks complicated, but in arithmetic we have

$$\neg\neg Q \leftrightarrow Q$$

for all quantifier-free formulas, and

$$\neg\neg\forall k \neg\neg B \leftrightarrow \forall k \neg\neg B$$

is provable intuitionistically. Therefore

$$A^N \leftrightarrow \forall k \neg\neg\exists n \forall m P(k,n,m).$$

and so

$$\mathrm{PA}^\omega \vdash \forall k \exists n \forall m P(k,n,m) \Rightarrow \mathrm{HA}^\omega \vdash \forall k \neg\neg\exists n \forall m P(k,n,m).$$

The classical functional interpretation

We cannot give a direct computational interpretation to classical arithmetic i.e. it is *not* the case that

$$\text{if } \mathrm{PA}^\omega \vdash A \text{ then } \mathrm{HA}^\omega \vdash \forall y A_D(t, y)$$

for some $t \in \mathrm{T}$. However, what we do have is:

1. A computational interpretation of Heyting arithmetic
2. An embedding of Peano arithmetic into Heyting arithmetic

So why not combine them? I.e.

$$\mathrm{PA}^\omega \mapsto \mathrm{HA}^\omega \mapsto \text{System } \mathsf{T}$$

Gödel's main theorem (second part)

Gödel's soundness theorem for *classical logic* says that we can translate a proof of A to a program witnessing $\exists x \forall y (A^N)_D(x, y)$.

Theorem (K. Gödel, 1958)

Suppose that

$$\mathrm{PA}^\omega \vdash A$$

Then there exists a term t of System T *such that*

$$\mathrm{HA}^\omega \vdash \forall y (A^N)_D(t, y)$$

and moreover, we can formally extract t from the proof of A.

Proof.

Combine the soundness theorem for intuitionistic logic with the negative translation. □

The classical functional interpretation of ∀∃∀ theorems

What is the functional interpretation of $B :\equiv \forall k \neg\neg \exists n \forall m P(k,n,m)$?

Recalling the interpretation of implication we have

$$\begin{aligned} \forall k \neg\neg \exists n \forall m P(k,n,m) &\mapsto \forall k \neg(\exists n \forall m P(k,n,m) \to \bot) \\ &\mapsto \forall k \neg \exists g \forall n \neg P(k,n,g(n)) \\ &\mapsto \forall k (\exists g \forall n \neg P(k,n,g(n)) \to \bot) \\ &\mapsto \exists \Phi \forall k, g P(k, \Phi(k,g), g(\Phi(k,g))) \end{aligned}$$

Therefore in the special case of theorems of this form, we have

$$\text{if PA}^\omega \vdash \forall k \exists n \forall m P(k,n,m) \text{ then HA}^\omega \vdash \forall k, g P(k, t(k,g), g(t(k,g)))$$

for some term t if System T.

We can equivalently view this as a bound i.e.

$$\mathrm{T} \vdash \forall k, g, \exists n \leq t(g,k) P(k,n,g(n))$$

We now see what was going on with the minimum principle!

Outline

The classical functional interpretation of ∀∃ statements

Suppose that $PA^{\omega} \vdash B$ where $B :\equiv \forall u \exists v Q(u, v)$. What does the classical functional interpretation do in this case?

Let's first look at the negative translation. We have

$$B^N \equiv \neg\neg(\forall u \exists v Q(u, v)) \equiv \neg\neg\forall u \neg\neg\exists v \neg\neg Q(u, v) \leftrightarrow \forall u \neg\neg\exists v Q(u, v)$$

where the equivalence $\leftrightarrow$ is possible in Heyting arithmetic. Therefore

$$\mathrm{HA} \vdash \forall u \neg\neg\exists v Q(u, v)$$

But what is the functional interpretation of this? We have

$$\begin{aligned} \forall u \neg\neg\exists v Q(u, v) &\mapsto \forall u \neg\exists v \neg Q(u, v) \\ &\mapsto \forall u \exists v \neg\neg Q(u, v) \\ &\mapsto \exists f \forall u Q(u, f(u)). \end{aligned}$$

But this is the same as the direct, intuitionistic functional interpretation!

Remark. What really going on here is that the functional interpretation admits Markov's principle $\neg\neg\exists x A_0(x) \to \exists x A_0(x)$ for any quantifier-free formula $A_0(x)$.

Program extraction theorem

Theorem

Suppose that

$$\mathrm{PA}^{\omega} \vdash \forall u \exists v Q(u,v).$$

Then there exists a term t of System T *such that*

$$\mathrm{HA}^{\omega} \vdash \forall u Q(u,tu)$$

and moreover, we can formally extract t from the proof of A.

In other words, for the special case of $\forall\exists$ theorems, we can extract a **direct** witness from their proof, even if their proof uses non-constructive reasoning and therefore doesn't seem to have any computational meaning.

At this point, you should have two burning questions:

- **How is the program extraction theorem even possible?**
- **Can we use it to find interesting computational results in classical mathematics?**

Outline

The computational interpretation of $\forall\exists\forall$ theorems

Theorem

There are of statements of the form $\forall a \exists x \forall y P(a,x,y)$ with

$$\mathrm{PA}^{\omega} \vdash \forall a \exists x \forall y P(a,x,y)$$

but such that there is no computable functional Φ satisfying

$$\mathcal{S}^{\omega} \models \forall a, y P(a, \Phi(a), y).$$

Proof. E.g. $\forall f \exists x \forall y (f(x) \leq f(y))$.

Theorem

Whenever

$$\mathrm{PA}^{\omega} \vdash \forall a \exists x \forall y P(a,x,y)$$

there exists a term t of System T such that

$$\mathrm{HA}^{\omega} \vdash \forall a, g P(a, t(a,g), g(t(a,g))).$$

Proof. Soundness of the negative translation + Dialectica interpretation.

Theorem

Whenever

$$\mathrm{PA}^\omega \vdash \forall u \exists v Q(u,v)$$

there exists a term s of System T such that

$$\mathrm{HA}^\omega \vdash \forall u Q(u, s(u)).$$

But how can this be consistent with the previous slide? What if we have a proof of the following form?

$$\forall a \exists x \forall y P(a,x,y) \to \forall u \exists v Q(u,v)$$

Back to least element principle

Theorem

For any function $f : \mathbb{N} \to \mathbb{N}$ *and* $u : \mathbb{N}$ *there exists some* $v : \mathbb{N}$ *such that*

$$f(v) \leq f(2v + u)$$

Proof.

By the least element principle there exists some x such that

$$\forall y (f(x) \leq f(y)).$$

Set $v := x$. □

But from the previous discussion

- x is in general non-computable.
- v is computable by a System T term t.
- t can be extracted from the above proof!

A proof theoretic analysis I

Fixing f as a parameter, our proof uses the implication

$$\exists x \forall y (f(x) \leq f(y)) \to \forall u \exists v (f(v) \leq f(2v+u))$$

The Dialectica interpretation of this is

$$\exists Y, V \forall u, x (f(x) \leq f(Yux) \to f(Vux) \leq f(2Vux+u))$$

This is easy! Just define

$$Vux := x \quad Yux = 2x+u$$

and we have

$$\forall x, u (f(x) \leq f(2x+u) \to f(x) \leq f(2x+u)).$$

A proof theoretic analysis II

Our proof also uses

$$\exists x \forall y (f(x) \leq f(y))$$

The negative translation + Dialectica interpretation of this is

$$\exists X \forall g (f(Xg) \leq f(g(Xg)))$$

Define

$$Xg := \begin{cases} 0 & \text{if } f(0) \leq f(g0) \\ g0 & \text{if } f(g(0)) \leq f(g^{(2)}(0)) \\ g^{(2)}(0) & \text{if } f(g^{(2)}(0)) \leq f(g^{(3)}(0)) \\ \dots & \dots \end{cases}$$

A proof theoretic analysis III

To summarise, we have

- $\exists X \forall g(f(Xg) \leq f(g(Xg)))$
- $\exists Y, V \forall u, x(f(x) \leq f(Yux) \rightarrow f(Vux) \leq f(2Vux + u))$

and we want

$$\exists h \forall u(f(hu) \leq f(2hu + u))$$

Just put everything together, and define $hu := Vxu$ for $x := Xg$ and $g := Yu$ i.e.

$$\begin{aligned} hu &= Vu(X(Yu)) \\ &= X(Yu) \\ &= X(\lambda x.(2x+u)) \quad = \begin{cases} 0 & \text{if } f(0) \leq f(u) \\ u & \text{if } f(u) \leq f(3u) \\ 3u & \text{if } f(3u) \leq f(7u) \\ \ldots & \ldots \end{cases} \end{aligned}$$

In the end we get $hu = (2^N - 1)u$ for some N.

Why it works in general

Suppose that a theorem $B :\equiv \forall u \exists v B(u, v)$ is proven using some nonconstructive lemma $A :\equiv \exists x \forall y A(x, y)$.

Naive idea. In order to find a function f satisfying $\forall u B(u, fu)$ we need to find some x satisfying $\forall y A(x, y)$. We cannot compute this x, therefore no computable f exists.

Recall the functional interpretation of implication:

$$(\exists x \forall y A(x, y) \to \forall u \exists v B(u, v)) \mapsto \exists V, Y \forall x, u(A(x, Yxu) \to B(u, Vxu))$$

Suppose we have functionals V, Y satisfying the interpretation of implication together with an indirect interpretation of $\exists x \forall y A(x, y)$ i.e. a functional Φ such that

$$(*) \quad \forall g A(\Phi g, g(\Phi g)).$$

For each u define the function $g_u : \mathbb{N} \to \mathbb{N}$ by $g_u(x) := Yxu$, and define

$$f(u) := V(\Phi g_u)u.$$

Then for any input u, by $(*)$ we have $A(\Phi g_u, g_u(\Phi g_u)) \equiv A(\Phi g_u, Y(\Phi g_u)u)$. Therefore $B(u, V(\Phi g_u)u) \equiv B(u, f(u))$ holds.

Theorem

Let R be a commutative ring. Suppose that r lies in the intersection of all prime ideals of R. Then r is nilpotent i.e. $\exists e > 0 (r^e = 0)$.

Proof.

Suppose that r is not nilpotent. Define

$$\Sigma := \{I \subset R \mid I \text{ is an ideal satisfying } \forall e > 0 (r^e \notin I)\}.$$

Then $\{0\} \in \Sigma$ (by our assumption), and Σ is chain-complete w.r.t. inclusion, so by Zorn's lemma it has a maximal element M.

We show that M is prime: If $m, n \notin M$ then $M + (m)$ and $M + (n)$ are proper extensions of M, so by maximality there exist $e_1, e_2 > 0$ such that $r^{e_1} \in M + (m)$ and $r^{e_2} \in M + (n)$. Therefore

$$r^{e_1 + e_2} \in M + (mn)$$

and so $M + (mn) \notin \Sigma$, which means that $mn \notin M$. Since $r^1 \notin M$, r cannot lie in the intersection of all prime ideals. □

Structure of proof

Roughly speaking, the theorem has the following form:

$$\forall R, r \in \bigcap P \left(\exists M \, \texttt{Maximal}_{\Sigma(R,r)}(M) \rightarrow \exists e > 0 (r^e = 0)\right)$$

Very roughly, this is partially interpreted as

$$\forall R, r \in \bigcap P, M \exists e, y \left(\texttt{ApproxMaximal}_{\Sigma(R,r)}(M, y) \rightarrow e > 0 \wedge r^e = 0\right)$$

Our proof contains a procedure which transforms:

A program for computing approximately maximal ideals $\mapsto$ A program for finding e

How do we compute approximations to maximal ideals?

- The functional interpretation would formally apply some kind of well-founded recursion on trees (following [Spector, 1962, Berardi et al., 1998]);
- People in constructive algorithm would design some kind of well-founded recursion over trees.

Question. How are these approaches related?

Outline

1. The Gödel-Gentzen negative translation
2. The extraction of programs for $\forall\exists$ theorems
3. The program extraction theorem
4. Proof mining: Bounds for existential statements
5. Focus: The finite convergence principle and metastability
6. Conclusion
7. References

The potential of program extraction

We have already seen some examples of witness extraction from $\forall\exists$ statements, our running example being

Theorem

There exists a function $X : \mathbb{N} \to \mathbb{N}$ such that for all n we have $X(n) \geq n$ and $X(n)$ prime.

But you don't need sophisticated proof theoretic techniques to be able to do this. So are there examples where the formal analysis of a proof can yield genuinely new numerical information from proofs?

The answer is an emphatic YES. This is the so-called 'proof mining' program.

Central to the success of proof mining program are the following phenomena:

- One can typically extract a witnesses for $\forall\exists$ statements even when the underlying proofs are non-constructive;
- Certain mathematical principles, particularly forms of *compactness*, do not contribute to the complexity of extracted bounds, leading to uniform polynomial bounds from proofs which employ heavy machinery from analysis.

A brief history of proof mining

- Pioneered by Kreisel in the 1950s, who proposed 'unwinding' constructive content from proofs using proof theoretic methods. Case studies in number theory and abstract algebra.
- In the 1980s, both Girard and Luckhardt carry out case studies and obtain bounds (van der Waerden's theorem and Roth's theorem respectively)
- From 1990s onwards, Kohlenbach finds numerous applications, in approximation theory and fixed point theory in particular. Proof mining takes off!
- In the 2010s Avigad, Towsner and others analyse convergence proofs in ergodic theory.
- In the last few years, Kohlenbach and his students find applications in convex optimization.
- **2018: Where to next?**

Example: Uniqueness of best approximation

Theorem

Let $n \in \mathbb{N}$ and $f \in C[0,1]$ be fixed. Let

$$dist(f, P_n) := \inf_{p \in P_n} \|f - p\|$$

where P_n is the space of all polynomials with degree $\leq n$. Then there exists a polynomial of best approximation i.e. a polynomial p^* such that

$$\|f - p^*\| = dist(f, P_n),$$

and moreover, this polynomial is unique i.e. for all $p_1, p_2 \in P_n$

$$\bigwedge_{i=1,2} (\|f - p_i\| = dist(f, P_n)) \to p_1 = p_2.$$

Let's look a bit more closely at uniqueness:

$$\forall n \in \mathbb{N} \forall f \in C[0,1] \forall p_1, p_2 \in P_n \left(\bigwedge_{i=1,2} (\|f - p_i\| = \mathrm{dist}(f, P_n)) \to p_1 = p_2 \right).$$

Now, equality $=$ over the real numbers is actually a $\forall$-statement and so written out fully, uniqueness becomes

$$\begin{cases} \forall n \in \mathbb{N} \forall f \in C[0,1] \forall p_1, p_2 \in P_n \\ \left(\forall j \bigwedge_{i=1,2} (\|f - p_i\| - \mathrm{dist}(f, P_n) < 2^{-j}) \to \forall k \, \|p_1 - p_2\| < 2^{-k} \right). \end{cases}$$

The (partial) functional interpretation of this is the following:

$$\begin{cases} \forall n, k \in \mathbb{N} \forall f \in C[0,1] \forall p_1, p_2 \in P_n \exists j \\ \left(\bigwedge_{i=1,2} (\|f - p_i\| - \mathrm{dist}(f, P_n) < 2^{-j}) \to \|p_1 - p_2\| < 2^{-k} \right). \end{cases}$$

A modulus of uniqueness

In the case of both the uniform norm and the L_1 norm, it is possible to extract a term Φ of System T such that

$$\begin{cases} \forall n, k \in \mathbb{N} \forall f \in C[0,1] \forall p_1, p_2 \in P_n \exists j \\ \left(\bigwedge_{i=1,2} (\|f - p_i\| - \mathrm{dist}(f, P_n) < 2^{-\Phi(f,n,k)}) \to \|p_1 - p_2\| < 2^{-k} \right). \end{cases}$$

where Φ is independent of p_1, p_2.

Remark. Φ is known as the modulus of uniqueness.

Explicit moduli of uniqueness are given in the following papers:

- de La Vallée Poussin's proof of uniqueness of best Chebychev approximation [Kohlenbach, 1993a];
- Young's proof of uniqueness of best Chebychev approximation [Kohlenbach, 1993b];
- Cheney's proof of uniqueness of best L_1 approximation [Kohlenbach and Oliva, 2003a].

In some cases these results even improved known results in the literature.

More recent work

For a comprehensive account of proof mining see [Kohlenbach, 2008] (the standard text on the subject).

Following early success in approximation theory, proof interpretations have been used to extract new quantitative information, and establish abstract generalizations, in the following areas in particular:

- Fixed point theory
- Ergodic theory
- Convex optimization

For individual expository articles see e.g.

- Kohlenbach, U. and Oliva, P. (2003b). A systematic way of analyzing proofs in mathematics.
  Proceedings of the Steklov Institute of Mathematics, 242:136–164
- Avigad, J. (2009). The metamathematics of ergodic theory.
  Annals of Pure and Applied Logic, 157:64–76
- Kohlenbach, U. Proof theoretic methods in nonlinear analysis.
  To appear in: Proc. Int. Cong. of Math. - ICM 2018

Outline

An important ∀∃∀ theorem

Theorem

Let (x_n) be a nondecreasing sequence of rational numbers in the unit interval $[0, 1]$. Then (x_n) converges to some limit.

What is the formal version of convergence? Naively, if (x_n) is a sequence in some space X, convergence in X means

$$\exists x \in X \forall k \exists n \forall m (|x_{n+m} - x| \leq 2^{-k})$$

But if we are in a complete space, we can instead use Cauchy convergence, which only refers to the sequence itself.

Theorem (Formal version.)

Let (x_n) be a nondecreasing sequence of rational numbers in $[0, 1]$. Then

$$\forall k \exists n \forall m (|x_{n+m} - x_n| \leq 2^{-k}).$$

The functional interpretation of convergence

Theorem (Functional interpretation)

Let (x_n) be a nondecreasing sequence of rational numbers in $[0, 1]$. Then there exists a function $N : \mathbb{N} \to \mathbb{N}$ such that

$$\forall k, m(|x_{N(k)+m} - x_{N(k)}| \leq 2^{-k}).$$

Remark. *The function N is a so-called modulus of convergence for (x_n).*

Example

- $x_n := 1 - \frac{1}{n}$ has modulus of convergence $N(k) =$

Even basic things like convergence are fundamentally non-computable

Theorem (E. Specker, 1949)

There exist computable, monotonically increasing, bounded sequences of rational numbers which do not have a computable modulus of convergence.

Note. Just sequences are known as Specker sequences.

Conclusion.

- There are simple, everyday mathematical facts which are fundamentally non-computable.
- Direct program extraction only works for proofs which don't use any law of excluded-middle.
- The vast majority of normal mathematical proofs are beyond program extraction...

But it's not quite as bad as it looks!

The monotone convergence theorem

Recall in the last lecture we discussed the *monotone convergence principle*:

Theorem

Let (x_n) *be a nondecreasing sequence of rational numbers in* $[0,1]$. *Then*

$$\forall k \exists n \forall m (|x_{n+m} - x_n| \leq 2^{-k}).$$

We learned that in general there is no computable $N : \mathbb{N} \to \mathbb{N}$ satisfying

$$\forall k, \forall m (|x_{N(k)+m} - x_{N(k)}| \leq 2^{-k}),$$

due to the result of Specker.

But we now have a procedure for dealing with non-computable statements like this.

Let's first take a look at a proof.

Proof.

Suppose that the monotone convergence principle fails i.e. there exists some k such that

$$\forall n \exists m (|x_{n+m} - x_n| > 2^{-k}).$$

Then there exists a function $g : \mathbb{N} \to \mathbb{N}$ such that

$$\forall n (|x_{n+g(n)} - x_n| > 2^{-k}).$$

Define the function $\tilde{g}(n) = n + g(n)$. Then we have a sequence

$$0 \leq x_0 < x_{\tilde{g}(0)} < x_{\tilde{g}^{(2)}(0)} < \ldots <$$

with $x_{\tilde{g}^{(i+1)}(0)} - x_{\tilde{g}^{(i)}(0)} > 2^{-k}$, therefore

$$x_{\tilde{g}^{(2^k)}} > 1$$

a contradiction. □

The computational content of the proof

Our proof gave us some indirect computational information, namely

$$\forall k, g \exists n \leq \tilde{g}^{(2^k)}(0)(|x_{n+g(n)} - x_n| \leq 2^{-k}),$$

or in other words

$$\forall k, g \exists n \leq \tilde{g}^{(2^k)}(0) \forall i, j \in [n, n + g(n)](|x_i - x_j| \leq 2^{-k})$$

Note that we can rephrase this statement entirely, so as only to refer to a finite part of (x_n). Let $M = \tilde{g}^{(2^k+1)}(0)$. We have the following:

Theorem (Finite convergence principle)

Let $k \in \mathbb{N}$, $g : \mathbb{N} \to \mathbb{N}$, and suppose that $0 \leq x_0 \leq x_1 \leq \ldots \leq x_M \leq 1$, where M is a sufficiently large number which depends only on k and g. Then there exists some $0 \leq n \leq n + g(n) \leq M$ such that $|x_i - x_j| \leq 2^{-k}$ for all $n \leq i, j \leq n + g(n)$.

This is the so-called *finite convergence principle*, made explicit by T. Tao's in

Tao, T. (2008a). Soft analysis, hard analysis, and the finite convergence principle. Essay, published as Ch. 1.3 of [Tao, 2008b], original version available online at `http://terrytao.wordpress.com/2007/05/23/soft-analysis-hard-analysis-and-the-finite-convergence-principle/`

- The finite convergence principle is not just an esoteric logical reformulation of a well-known concept. It is actually used in mathematics in e.g. the proof of the Szemerédi regularity lemma.
- In his essay, Tao draws attention to the fact that many infinitary ('soft', qualitative') statements have finitary ('hard', 'quantitative') analogous, which have useful applications.
- It was later observed that this correspondence between soft and hard statements is just the classical functional interpretation!

Idea. Proof interpretations do much more that just extracting numerical information. They help us understand and formalize the connection between infinitary and finintary statements in mathematics.

Convergence principles are widely studied in proof mining. Here, the functional which witnesses the corresponding finitary principle is knows as a rate of metastability.

Too see the functional interpretation applied to obtain finitary versions of other infinitary principles see e.g.

- Gaspar, J. and Kohlenbach, U. (2010). On Tao's "finitary" infinite pigeonhole principle.
  Journal of Symbolic Logic, 75(1):355–371
- Safarik, P. and Kohlenbach, U. (2010). On the interpretation of the Bolzano-Weierstrass principle.
  Mathematical Logic Quarterly, 56(5):508–532
- Powell, T. (2018). Well quasi-orders and the functional interpretation.
  To appear in: Schuster, P., Seisenberger, M. and Weiermann, A. editors, *Well Quasi-Orders in Computation, Logic, Language and Reasoning*, Trends in Logic, Springer

Outline

How are proof theoretic tools applied to new areas?

Key steps:

1. Are there theorems in this area which have the right logical structure? What kind of information could I hope to extract?
2. How do I formalize the proofs? How do I represent the underlying spaces?
3. Analyse some concrete proofs.
4. What is going on more generally? Can these proofs be expressed in an abstract logical framework?
5. Develop new metatheorems which *guarantee* that, under certain conditions, programs can be extracted.

Potential new areas:

- Number theory?
- Probability theory?
- Financial mathematics?

Outline

References I

Avigad, J. (2009).
The metamathematics of ergodic theory.
Annals of Pure and Applied Logic, 157:64–76.

Berardi, S., Bezem, M., and Coquand, T. (1998).
On the computational content of the axiom of choice.
Journal of Symbolic Logic, 63(2):600–622.

Gaspar, J. and Kohlenbach, U. (2010).
On Tao's "finitary" infinite pigeonhole principle.
Journal of Symbolic Logic, 75(1):355–371.

Kohlenbach, K. and Oliva, P. (2003a).
Proof mining in the L_1-approximation.
Annals of Pure and Applied Logic, 121:1–38.

Kohlenbach, U.
Proof theoretic methods in nonlinear analysis.
To appear in: Proc. Int. Cong. of Math. - ICM 2018.

References II

Kohlenbach, U. (1993a).
Effective moduli from ineffective uniqueness proofs. an unwinding of de la vallée poussin's proof for chebycheff approximation.
Annals of Pure and Applied Logic, 64:27–94.

Kohlenbach, U. (1993b).
New effective moduli of uniqueness and uniform a-priori estimates for constants of strong unicity by logical analysis of known proofs in best approximation theory.
Numer. Funct. Anal. Optim., 14:581–606.

Kohlenbach, U. (2008).
Applied Proof Theory - Proof Interpretations and their Use in Mathematics.
Springer Monographs in Mathematics. Springer.

Kohlenbach, U. and Oliva, P. (2003b).
A systematic way of analyzing proofs in mathematics.
Proceedings of the Steklov Institute of Mathematics, 242:136–164.

Powell, T. (2018).
Well quasi-orders and the functional interpretation.
To appear in: Schuster, P., Seisenberger, M. and Weiermann, A. editors, *Well Quasi-Orders in Computation, Logic, Language and Reasoning*, Trends in Logic, Springer.

References III

Safarik, P. and Kohlenbach, U. (2010).
On the interpretation of the Bolzano-Weierstrass principle.
Mathematical Logic Quarterly, 56(5):508–532.

Spector, C. (1962).
Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles in current intuitionistic mathematics.
In Dekker, F. D. E., editor, *Recursive Function Theory: Proc. Symposia in Pure Mathematics*, volume 5, pages 1–27. American Mathematical Society, Providence, Rhode Island.

Tao, T. (2008a).
Soft analysis, hard analysis, and the finite convergence principle.
Essay, published as Ch. 1.3 of [Tao, 2008b], original version available online at `http://terrytao.wordpress.com/2007/05/23/soft-analysis-hard-analysis-and-the-finite-convergence-principle/`.

Tao, T. (2008b).
Structure and Randomness: Pages from Year 1 of a Mathematical Blog.
American Mathematical Society.