

Three views of LPO and LLPO

Jeffry Hirst
Appalachian State University
Boone, NC

April 18, 2022

Midwest Computability Seminar

LPO: Limited Principle of Omniscience

LPO: If $f : \mathbb{N} \rightarrow \{0, 1\}$ then 0 is in its range or it isn't.

Written as an $\forall\exists$ formula:

$$\forall f \exists n (n = 0 \vee n = 1 \wedge (n = 0 \leftrightarrow \exists t (f(t) = 0)))$$

A (the) realizer for LPO:

$$R_{\text{LPO}}(f) = \begin{cases} 0 & \text{if } 0 \in \text{Range}(f) \\ 1 & \text{if } 0 \notin \text{Range}(f) \end{cases}$$

LLPO: Lesser Limited Principle of Omniscience

LLPO: If $f : \mathbb{N} \rightarrow \{0, 1\}$ and $0 \in \text{Range}(f)$, then the first 0 occurs at an even integer or at an odd integer.

Written as an $\forall\exists$ formula:

$$\forall f \exists n (\exists t (f(t) = 0) \rightarrow n \equiv_{\text{mod } 2} \mu t (f(t) = 0))$$

Values of a realizer:

<table><tr><td>n</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>...</td></tr><tr><td>$f(n)$</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>...</td></tr></table>	n	0	1	2	3	4	...	$f(n)$	1	1	0	1	0	...	$R_{\text{LLPO}}(f) = 0$
n	0	1	2	3	4	...									
$f(n)$	1	1	0	1	0	...									
<table><tr><td>n</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>...</td></tr><tr><td>$g(n)$</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>...</td></tr></table>	n	0	1	2	3	4	...	$g(n)$	1	1	1	0	0	...	$R_{\text{LLPO}}(g) = 1$
n	0	1	2	3	4	...									
$g(n)$	1	1	1	0	0	...									
<table><tr><td>n</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>...</td></tr><tr><td>$h(n)$</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>...</td></tr></table>	n	0	1	2	3	4	...	$h(n)$	1	1	1	1	1	...	$R_{\text{LLPO}}(h) = 0 \text{ or } 1$
n	0	1	2	3	4	...									
$h(n)$	1	1	1	1	1	...									

Weihrauch reductions

Suppose P and Q are problems.

We say P is Weihrauch reducible to Q (and write $P \leq_w Q$) if there are (partial) computable functionals Φ and Ψ such that if p is an instance of P , then

- $\Phi(p)$ is an instance of Q and
- for any solution s of $\Phi(p)$, $\Psi(s, p)$ is a solution of p .

That is, if R_Q is any realizer of Q , then $\Psi(R_Q(\Phi(p)), p)$ is a realizer for P .

If Ψ does not use the original problem p , we say P is strongly Weihrauch reducible to Q .

Weihrauch reductions

If R_Q is any realizer of Q , then $\Psi(R_Q(\Phi(p)), p)$ is a realizer for P .

- Φ is a pre-processor, turning P problems into Q problems.
- Ψ is a post-processor, turning Q solutions (with copies of P problems) into P solutions.
- Φ and Ψ uniformly turn any Q realizer into a P realizer.

Example of a Weihrauch reduction

We will show that $\text{LLPO} \leq_W \text{LPO}$, relying on post-processing.

Instructions for Φ :

Input an LLPO problem p .

Do nothing and output p .

Instructions for Ψ :

Input p and $R_{\text{LPO}}(\Phi(p))$.

Note: $R_{\text{LPO}}(\Phi(p))$ is the LPO solution for p .

If $R_{\text{LPO}}(p) = 1$ (so p is all 1s)

then output 1 and halt.

If $R_{\text{LPO}}(p) = 0$ (so p contains a 0)

then find the first 0 and output the location (mod 2).

Ψ is partial. A false value for $R_{\text{LPO}}(\Phi(p))$ may loop.

Example of a Weihrauch reduction

We will show that $\text{LLPO} \leq_W \text{LPO}$, relying on pre-processing.

Instructions for Φ :

Input an LLPO problem p .

For each n , define $q(n)$ by

if there is an $m \leq n$ such that $p(m) = 0$ and the first such m is even, then set $q(n) = 0$, and set $q(n) = 1$ otherwise.

Instructions for Ψ :

Input p and $R_{\text{LPO}}(\Phi(p))$.

Output $R_{\text{LPO}}(\Phi(p))$.

Φ and Ψ are total. Ψ doesn't use p , so $\text{LLPO} \leq_{\text{sw}} \text{LPO}$.

Parallelization and Weihrauch reducibility

For a Weihrauch problem P , the parallelization is denoted by \widehat{P} .

\widehat{P} accepts a sequence of P problems as input, and outputs the sequence of their solutions.

For example, written as an $\forall\exists$ formula, $\widehat{\text{LPO}}$ is

$$\forall\langle f_i \rangle \exists\langle n_i \rangle \forall i (n_i = 0 \vee n_i = 1 \wedge (n_i = 0 \leftrightarrow \exists t (f_i(t) = 0)))$$

which is a set existence statement.

Reverse mathematics: The base theory

Reverse Mathematics measures the strength of theorems by proving equivalence results over...

The base theory RCA_0 :

Variables for natural numbers and sets of natural numbers

Axioms

Arithmetic axioms

(e.g. $n + 0 = n$ and $n + (m') = (n + m)'$)

Induction for particularly simple formulas

Recursive comprehension:

If you can compute a set, then it exists.

Reverse mathematics: The big five

Many theorems of mathematics are equivalent to one of four statements (over the base theory RCA_0). [3]

$$\text{RCA}_0 < \text{WKL}_0 < \text{ACA}_0 < \text{ATR}_0 < \Pi_1^1\text{-CA}_0$$

Where does $\widehat{\text{LPO}}$ fit?

Reverse mathematics: The big five

Many theorems of mathematics are equivalent to one of four statements (over the base theory RCA_0). [3]

$$\text{RCA}_0 < \text{WKL}_0 < \text{ACA}_0 < \text{ATR}_0 < \Pi_1^1\text{-CA}_0$$

Where does $\widehat{\text{LPO}}$ fit?

Prop (RCA_0): The following are equivalent:

- (1) ACA_0 .
- (2) $\widehat{\text{LPO}}$.

For the reversal, find the range of an arbitrary injection on \mathbb{N} .

Reverse mathematics: The big five

Many theorems of mathematics are equivalent to one of four statements (over the base theory RCA_0). [3]

$$\text{RCA}_0 < \text{WKL}_0 < \text{ACA}_0 < \text{ATR}_0 < \Pi_1^1\text{-CA}_0$$

Where does $\widehat{\text{LLPO}}$ fit?

Reverse mathematics: The big five

Many theorems of mathematics are equivalent to one of four statements (over the base theory RCA_0). [3]

$$\text{RCA}_0 < \text{WKL}_0 < \text{ACA}_0 < \text{ATR}_0 < \Pi_1^1\text{-CA}_0$$

Where does $\widehat{\text{LLPO}}$ fit?

Prop (RCA_0): The following are equivalent:

- (1) WKL_0 .
- (2) $\widehat{\text{LLPO}}$.

For the reversal, separate the ranges of two injections with disjoint ranges.

Computability theoretic observations

There is a computable injection on \mathbb{N} with a range that computes $0'$.

There is a computable $\widehat{\text{LPO}}$ problem such that every solution computes $0'$.

There is an ω model of WKL_0 containing only low sets.

Every computable $\widehat{\text{LLPO}}$ problem has a low solution.

Thus, $\widehat{\text{LPO}} \not\leq_w \widehat{\text{LLPO}}$

Computability theoretic observations

There is a computable injection on \mathbb{N} with a range that computes $0'$.

There is a computable $\widehat{\text{LPO}}$ problem such that every solution computes $0'$.

There is an ω model of WKL_0 containing only low sets.

Every computable $\widehat{\text{LLPO}}$ problem has a low solution.

Thus, $\widehat{\text{LPO}} \not\leq_W \widehat{\text{LLPO}}$, and so $\text{LPO} \not\leq_W \text{LLPO}$.

Higher order reverse mathematics

Work in collaboration with Carl Mummert.

Kohlenbach [5] proposed an extension of the axiom systems of reverse mathematics to all finite types.

In this setting, we can prove equivalences between Skolemized functional existence statements.

As an example,

$$\text{LPO} : \forall f \exists n (n = 0 \vee n = 1 \wedge (n = 0 \leftrightarrow \exists t (f(t) = 0)))$$

$$\begin{aligned} (\text{LPO}) : \exists R_{\text{LPO}} \forall f (R_{\text{LPO}}(f) = 0 \vee R_{\text{LPO}}(f) = 1 \wedge \\ (R_{\text{LPO}}(f) = 0 \leftrightarrow \exists t (f(t) = 0))) \end{aligned}$$

Higher order RM: The base theory

Kohlenbach's [5] RCA_0^ω includes functionals of higher type, like $f : 2^{\mathbb{N}} \rightarrow \mathbb{N}$ and $g : 2^{\mathbb{N}} \rightarrow 2^{\mathbb{N}}$.

It includes:

- Restricted induction
- Primitive recursion (on \mathbb{N} with parameters)
- λ -abstraction

Naïvely, if you can compute a functional, it exists.

(LPO) as an ACA_0 analog

$$(LPO) : \exists R_{LPO} \forall f (R_{LPO}(f) = 0 \vee R_{LPO}(f) = 1 \wedge \\ (R_{LPO}(f) = 0 \leftrightarrow \exists t (f(t) = 0)))$$

(LPO) is Kohlenbach's (\exists^2) . The functional R_{LPO} is Kleene's E^2 .

$RCA_0^\omega + (LPO)$ is a conservative extension of ACA_0 for Π_2^1 formulas.[6]

And for (LLPO),

(LPO) as an ACA_0 analog

$$(LPO) : \exists R_{LPO} \forall f (R_{LPO}(f) = 0 \vee R_{LPO}(f) = 1 \wedge \\ (R_{LPO}(f) = 0 \leftrightarrow \exists t (f(t) = 0)))$$

(LPO) is Kohlenbach's (\exists^2) . The functional R_{LPO} is Kleene's E^2 .

$RCA_0^\omega + (LPO)$ is a conservative extension of ACA_0 for Π_2^1 formulas.[6]

And for (LLPO), there is a surprise:

Prop: (RCA_0^ω) The following are equivalent:

- (1) (LPO)
- (2) (LLPO)

$\text{RCA}_0^\omega \vdash (\text{LLPO}) \rightarrow (\text{LPO})$

Proof sketch: Working in RCA_0^ω , suppose we have R_{LLPO} .

Let $f = \langle 1, 1, 1, \dots \rangle$, and suppose $R_{\text{LLPO}}(f) = 0$.

Define a sequence of inputs:

$$g_n(m) = \begin{cases} 1 & \text{if } m \neq 2n + 1 \\ 0 & \text{if } m = 2n + 1 \end{cases}$$

For every n , $R_{\text{LLPO}}(g_n) = 1$, so

$$\lim_n R_{\text{LLPO}}(g_n) = 1 \neq 0 = R_{\text{LLPO}}(f) = R_{\text{LLPO}}(\lim_n g_n)$$

and R_{LLPO} is effectively sequentially discontinuous.

Apply Prop. 3.7 of Kohlenbach [5] to obtain (LPO).

If $R_{\text{LLPO}}(f) = 1$, revise the definition of g_n .

Grilliot's trick and Kohlenbach's proposition

We can replace the use of Prop. 3.7 of Kohlenbach [5] with part of the proof of Lemma 1 of Grilliot [4].

Let f and g_n be as before. Define the functional $J : 2^{\mathbb{N}} \rightarrow 2^{\mathbb{N}}$ for $h : \mathbb{N} \rightarrow 2$ and $j \in \mathbb{N}$ by

$$J(h)(j) = \begin{cases} 1 & \text{if } \forall x \leq j (h(x) \neq 0), \\ g_j(j) & \text{if } i \leq j \wedge i = \mu t (h(t) = 0). \end{cases}$$

Note that if $h = f$, then $J(h) = f$. If i is the least value such that $h(i) = 0$, then $J(h) = g_i$.

So h contains a zero if and only if $R_{\text{LLPO}}(J(h)) \neq R_{\text{LLPO}}(f)$. We can use R_{LLPO} and J to compute R_{LPO} .

The underlying computability theory

Why does $\text{RCA}_0^\omega \vdash (\text{LLPO}) \rightarrow (\text{LPO})$ when $\text{LPO} \not\leq_W \text{LLPO}$?

The two approaches yield different information.

$\text{LPO} \not\leq_W \text{LLPO}$ because there are no fixed computable pre-processing and post-processing functionals that can (uniformly) convert every realizer for LLPO into a realizer for LPO.

$\text{RCA}_0^\omega \vdash (\text{LLPO}) \rightarrow (\text{LPO})$ because given any realizer for LLPO we can compute a realizer for LPO. In the proof, when we said “suppose $R_{\text{LLPO}}(f) = 0$ ” we made a non-uniform choice.

What about the other half of the equivalence?

Prop: $\text{RCA}_0^\omega \vdash (\text{LPO}) \rightarrow (\text{LLPO})$.

The idea of the proof:

RCA_0^ω can prove the existence of the pre-processing and post-processing functionals in our second proof of $\text{LLPO} \leq_W \text{LPO}$. The implication follows by composition of functionals.

The post-processing functional in the first proof was not total, so that argument does not work in the higher order reverse mathematics setting.

Three questions, one answer.

- Are all Skolemized functional existence statements (SFEs) corresponding to WKL_0 statements equivalent to (LPO) in higher order reverse mathematics?

Three questions, one answer.

- Are all Skolemized functional existence statements (SFEs) corresponding to WKL_0 statements equivalent to (LPO) in higher order reverse mathematics?

No. Kohlenbach [5] noted statements about moduli of uniform continuity that are conservative over WKL_0 .

Three questions, one answer.

- Are all Skolemized functional existence statements (SFEs) corresponding to WKL_0 statements equivalent to (LPO) in higher order reverse mathematics?

No. Kohlenbach [5] noted statements about moduli of uniform continuity that are conservative over WKL_0 .

- Which SFEs corresponding to WKL_0 are equivalent to (LPO)? Are the others weaker and equivalent to each other, or is there a low-level higher-order zoo?

Three questions, one answer.

- Are all Skolemized functional existence statements (SFEs) corresponding to WKL_0 statements equivalent to (LPO) in higher order reverse mathematics?

No. Kohlenbach [5] noted statements about moduli of uniform continuity that are conservative over WKL_0 .

- Which SFEs corresponding to WKL_0 are equivalent to (LPO)? Are the others weaker and equivalent to each other, or is there a low-level higher-order zoo?
- What about SFEs corresponding to $\Pi_1^1\text{-CA}_0$ (and the Suslin functional) as compared to those corresponding to theorems equivalent to ATR_0 ?

References

- [1] Vasco Brattka and Guido Gherardi, *Effective choice and boundedness principles in computable analysis*, Bull. Symbolic Logic **17** (2011), no. 1, 73–117.
DOI [10.2178/bsl/1294186663](https://doi.org/10.2178/bsl/1294186663). MR2760117
- [2] Vasco Brattka, Guido Gherardi, and Arno Pauly, *Weihrauch Complexity in Computable Analysis* (2018), 50+11. arXiv:1707.03202v4
DOI [10.48550/arXiv.1707.03202](https://doi.org/10.48550/arXiv.1707.03202).
- [3] Harvey Friedman, *Abstracts: Systems of second order arithmetic with restricted induction, I and II*, J. Symbolic Logic **41** (1976), 557–559.
DOI [10.2307/2272259](https://doi.org/10.2307/2272259).
- [4] Thomas J. Grilliot, *On effectively discontinuous type-2 objects*, J. Symbolic Logic **36** (1971), 245–248. DOI [10.2307/2270259](https://doi.org/10.2307/2270259). MR290972
- [5] Ulrich Kohlenbach, *Higher order reverse mathematics*, Reverse mathematics 2001, Lect. Notes Log., vol. 21, Assoc. Symbol. Logic, La Jolla, CA, 2005, pp. 281–295.
DOI [10.1017/9781316755846](https://doi.org/10.1017/9781316755846). MR2185441
- [6] Nobuyuki Sakamoto and Takeshi Yamazaki, *Uniform versions of some axioms of second order arithmetic*, MLQ Math. Log. Q. **50** (2004), no. 6, 587–593.
DOI [10.1002/malq.200310122](https://doi.org/10.1002/malq.200310122). MR2096172
- [7] Stephen G. Simpson, *Subsystems of second order arithmetic*, 2nd ed., Perspectives in Logic, Cambridge University Press and ASL, 2009.
DOI [10.1017/CBO9780511581007](https://doi.org/10.1017/CBO9780511581007). MR2517689