

Leaf Management

Jeffrey L. Hirst

December 23, 2018

Abstract

Finding the set of leaves for an unbounded tree is a nontrivial process in both the Weihrauch and reverse mathematics settings. Despite this, many combinatorial principles for trees are equivalent to their restrictions to trees with leaf sets. For example, let $\widehat{\text{WF}}$ denote the problem of choosing which trees in a sequence are well-founded, and let $\widehat{\text{PK}}$ denote the problem of finding the perfect kernel of a tree. Let $\widehat{\text{WF}}_L$ and $\widehat{\text{PK}}_L$ denote the restrictions of these principles to trees with leaf sets. Then $\widehat{\text{WF}}$, $\widehat{\text{WF}}_L$, $\widehat{\text{PK}}$, and $\widehat{\text{PK}}_L$ are all equivalent to $\Pi_1^1\text{-CA}_0$ over RCA_0 , and all strongly Weihrauch equivalent.

Introduction

The first section of this paper shows that for unbounded trees, finding leaf sets is a nontrivial process. The second section describes an algorithm for transforming trees into trees with leaf sets in such a way that properties related to infinite paths and perfect subtrees are preserved. The main equivalence results are presented in this section. The paper closes with a section containing an application to hypergraphs, where the use of a combinatorial principle restricted to sequences of trees with leaf sets is central to the proof of a Weihrauch equivalence.

All relevant background information on reverse mathematics can be found in Simpson's text [7]. For background on Weihrauch analysis, see the work of Brattka, Gherardi, and Pauly [3].

Leaf sets

In second order arithmetic settings, a tree is encoded by a set of finite sequences of natural numbers that is closed under initial subsequences. For

any finite sequence σ , let $|\sigma|$ denote the length of σ . A leaf in a tree is a sequence that has no extensions in the tree. For a tree T , let $\text{leaf}(T)$ denote the set of leaves of T . A function $b : \mathbb{N} \rightarrow \mathbb{N}$ is a bounding function for T if for every $\sigma \in T$ and for every $i < |\sigma|$, $\sigma(i) \leq b(i)$. If a tree T has a bounding function, little set comprehension is required to calculate $\text{leaf}(T)$.

Proposition 1. (RCA_0) *If b is a bounding function for the tree T , then $\text{leaf}(T)$ exists.*

Proof. Working in RCA_0 , the sequence σ is a leaf of T if and only if $\sigma \in T$ and for each $j \leq b(|\sigma| + 1)$ we have $\sigma \hat{\ } j \notin T$. Thus, the set $\text{leaf}(T)$ is computable using T and b as parameters, and exists by recursive comprehension. \square

Let bleaf denote the Weihrauch problem that accepts a tree T and a bounding function b as inputs and outputs the set $\text{leaf}(T)$. For a bounded tree, the preceding proof describes a process for computing the leaf set. Consequently, bleaf is at the lowest level of the strong Weihrauch hierarchy, as stated in the following proposition.

Proposition 2. $\text{bleaf} \equiv_{\text{sW}} 0$.

Finding leaf sets for trees without bounding functions is nontrivial. The formulation of LPO parallelized in the next proposition is the one preceding Theorem 6.7 of Brattka, Gherardi, and Pauly [3], and the Boolean negation of Definition 2.6 of Brattka and Gherardi [2].

Proposition 3. (RCA_0) *The following are equivalent.*

- (1) ACA_0 .
- (2) *For every tree T , the set $\text{leaf}(T)$ exists.*
- (3) $\widehat{\text{LPO}}$: *If $\langle p_i \rangle_{i \in \mathbb{N}}$ is a sequence of sequences of natural numbers, then there is a function $z : \mathbb{N} \rightarrow \{0, 1\}$ such that for each i , $z(i) = 1$ if and only if $\exists n(p_i(n) = 0)$.*

Proof. We will work in RCA_0 throughout the proof. To see that (1) implies (2), note that $\sigma \in \text{leaf}(T)$ if and only if $\sigma \in T$ and $\forall j(\sigma \hat{\ } j \notin T)$. Thus arithmetical comprehension proves the existence of $\text{leaf}(T)$.

To see that (2) implies (3), assume (2) and let $\langle p_i \rangle_{i \in \mathbb{N}}$ be an instance of $\widehat{\text{LPO}}$. Consider the tree T constructed from $\langle p_i \rangle_{i \in \mathbb{N}}$ as follows. Every finite

sequence of ones is in T . For each $n \in \mathbb{N}$ the sequence σ_n consisting of $n + 1$ ones followed by a zero is in T . The sequence $\sigma_n \widehat{j}$ is in T if and only if $p_n(j) = 0$ and $\forall i < j (p_n(i) \neq 0)$. The set of sequences T exists by recursive comprehension and is a tree because it is closed under initial segments. Apply (2) to find $\text{leaf}(T)$. The function $z : \mathbb{N} \rightarrow \{0, 1\}$ defined by $z(n) = 1$ if and only if $\sigma_n \notin \widehat{\text{leaf}(T)}$ exists by recursive comprehension and is a solution of the instance of $\widehat{\text{LPO}}$.

To complete the proof, by Lemma III.1.3 of Simpson [7], it suffices to use (3) to find the range of an injection. Let $f : \mathbb{N} \rightarrow \mathbb{N}$ be an injection. By recursive comprehension, we can find the sequence $\langle p_i \rangle_{i \in \mathbb{N}}$ defined by $p_i(n) = 0$ if $f(n) = i$, and $p_i(n) = 1$ otherwise. Apply (3) to find z such that $z(i) = 1$ if and only if $\exists n (p_i(n) = 0)$. Then the set $\{i \mid z(i) = 1\}$ is the range of f , is computable from z , and so exists by recursive comprehension. \square

By virtue of our circuitous reverse mathematics treatment, we can easily prove the related Weihrauch reducibility result. Let leaf denote the problem that accepts a tree T as input and outputs the leaf set $\text{leaf}(T)$.

Proposition 4. $\text{leaf} \equiv_{\text{sW}} \widehat{\text{LPO}}$.

Proof. The proof that (2) implies (3) for Proposition 3 also shows that $\widehat{\text{LPO}} \leq_{\text{sW}} \text{leaf}$. To prove the reverse relation, fix T and let $\langle \sigma_i \rangle_{i \in \mathbb{N}}$ be an enumeration of the sequences in T . For each i , let $p_i(j) = 0$ if $\sigma_i \widehat{j} \in T$ and let $p_i(j) = 1$ otherwise. If z is a solution to this instance of $\widehat{\text{LPO}}$, then it is also a characteristic function for $\text{leaf}(T)$. \square

Transforming trees

As shown in the previous section, finding the leaf set for an arbitrary tree is a nontrivial process in both the reverse mathematics and Weihrauch settings. However, in many cases it is possible to uniformly transform trees into trees with leaf sets while preserving many Weihrauch equivalences and equivalence theorems of reverse mathematics. The transformation can be defined using the following operations on finite sequences. For every $\sigma \in \mathbb{N}^{<\mathbb{N}}$, let $\sigma + 1$ denote the sequence with exactly the same length as σ such that for all $n < |\sigma|$, $(\sigma + 1)(n) = \sigma(n) + 1$. For example, $\langle 1, 3, 5 \rangle + 1 = \langle 2, 4, 6 \rangle$. Similarly, define $\sigma \dot{-} 1$ so that $(\sigma \dot{-} 1)(n) = \sigma(n) \dot{-} 1$. Our main tree transformation is T^* as defined in the following theorem. Naïvely, T^* is created by adding 1 to every node of T and attaching a leaf labeled 0 to each positive node.

Lemma 5. (RCA_0) *Suppose $T \subset \mathbb{N}^{<\mathbb{N}}$ is a tree. The following are also trees: $T^- = \{\sigma \dot{-} 1 \mid \sigma \in T\}$, $T^+ = \{\sigma + 1 \mid \sigma \in T\}$, and $T^* = T^+ \cup \{\sigma \frown 0 \mid \sigma \in T^+\}$. Furthermore, given a sequence $\langle T_i \rangle_{i \in \mathbb{N}}$, we can find the sequence $\langle T_i^*, \text{leaf}(T_i^*) \rangle$.*

Proof. Working in RCA_0 , it is easy to use recursive comprehension to prove the existence of the sets T^- , T^+ , and T^* . The initial segments of the shifts $\sigma \dot{-} 1$ and $\sigma + 1$ are the shifts of initial segments of σ , so T^- and T^+ are trees. A proper initial segment of a sequence in the set $\{\sigma \frown 0 \mid \sigma \in T^+\}$ is an initial segment of an element of T^+ , so T^* is also a tree.

To complete the proof, suppose $\langle T_i \rangle_{i \in \mathbb{N}}$ is a sequence of trees. For each i , $\sigma \in T_i^*$ if and only if the last element of σ is positive and $\sigma \dot{-} 1 \in T_i$, or if $\sigma = \tau \frown 0$ and $\tau \dot{-} 1 \in T_i$. Thus recursive comprehension implies that $\langle T_i^* \rangle_{i \in \mathbb{N}}$ exists. For each i , the sequence $\sigma \in \text{leaf}(T_i^*)$ if and only if $\sigma \in T_i^*$ and the last entry of σ is 0. Thus RCA_0 can prove that the sequence of pairs $\langle T_i^*, \text{leaf}(T_i^*) \rangle$ exists. \square

The trees T and T^* share many properties. Information about paths and subtrees of one can be uniformly transformed to information about the other. As described in Simpson [7, Definition I.6.6], a subtree S of T is perfect if every sequence in T has incompatible extensions in T . The perfect kernel of T is the union of all the perfect subtrees of T .

Theorem 6. (RCA_0) *A tree T and the transform T^* satisfy the following.*

- (1) *T is well-founded if and only if T^* is well-founded.*
- (2) *T has at most one path if and only if T^* has at most one path.*
- (3) *S is a perfect subtree of T if and only if S^+ is a perfect subtree of T^* .*
- (4) *K is the perfect kernel of T if and only if K^+ is the perfect kernel of T^* .*

Proof. Each part follows from the fact that the map taking σ to $\sigma + 1$ is a bijection between the paths of T and those of T^* and also between the perfect subtrees of T and those of T^* . \square

The next two theorems list familiar equivalences for tree statements that continue to hold when restricted to trees with leaf sets. For both proofs, the central tool is the transformation from T to T^* . In the following theorem,

the labels used for the combinatorial principles are consistent with those used for the associated Weihrauch problems by Kihara, Marcone, and Pauly [6].

Theorem 7. (RCA_0) *The following are equivalent.*

- (1) ATR_0 .
- (2) The Σ_1^1 separation principle: *For any Σ_1^1 formulas $\varphi_0(n)$ and $\varphi_1(n)$ containing no free occurrences of Z , if $\neg\exists n(\varphi_0(n) \wedge \varphi_1(n))$, then*

$$\exists Z \forall n((\varphi_0(n) \rightarrow n \in Z) \wedge (\varphi_1(n) \rightarrow n \notin Z)).$$

- (3) $\Sigma_1^1\text{-SEP}$: *If $\langle T_{0,i} \rangle_{i \in \mathbb{N}}$ and $\langle T_{1,i} \rangle_{i \in \mathbb{N}}$ are sequences of trees such that for each i , at most one of $T_{0,i}$ and $T_{1,i}$ has an infinite path, then there is a set Z such that for all n , $T_{0,n}$ has an infinite path implies $n \in Z$ and $T_{1,n}$ has an infinite path implies $n \notin Z$.*
- (4) $\Sigma_1^1\text{-SEP}_L$: *Item (3) for $\langle T_{0,i}, \text{leaf}(T_{0,i}) \rangle_{i \in \mathbb{N}}$ and $\langle T_{1,i}, \text{leaf}(T_{1,i}) \rangle_{i \in \mathbb{N}}$, sequences of trees with leaf sets.*
- (5) $\Sigma_1^1\text{-CA}^-$: *If $\langle T_i \rangle_{i \in \mathbb{N}}$ is a sequence of trees each with at most one infinite path, then there is a set Z such that for all n , $n \in Z$ if and only if T_n has an infinite path.*
- (6) $\Sigma_1^1\text{-CA}_L^-$: *Item 5 for sequences of trees with leaf sets.*
- (7) PTT_1 : *If T has uncountably many paths then T has a non-empty perfect subtree.*
- (8) PTT_{1L} : *Item 7 for trees with leaf sets.*

Proof. The equivalence of (1) and (2) is Theorem V.5.1 of Simpson [7]. The existence of an infinite path in a tree can be written as a Σ_1^1 formula, so (2) implies (3). To prove the converse, use a bootstrapping argument, proving ACA_0 from (3) by creating a sequence of pairs of linear trees that compute the range of an injection. Then use ACA_0 and (3) to derive (2) by an application of Lemma 3.14 of Friedman and Hirst [5]. The equivalence of (5) and (1) is Theorem V.5.2 of Simpson [7], and the equivalence of (7) and (1) is Theorem V.5.5 of Simpson [7]. Item (4) is a restriction of (3), so (3) implies (4) trivially. The converse is an immediate consequence of Theorem 6. Similarly, (5) and (6) are equivalent, as are (7) and (8). \square

To avoid confusion with the subsystem $\Pi_1^1\text{-CA}_0$, in following theorem we use $\widehat{\text{WF}}$ as a label for the combinatorial principle denoted by $\Pi_1^1\text{-CA}$ in the article of Kihara, Marcone, and Pauly [6]. Note that $\widehat{\text{WF}}$ is the infinite parallelization of the the principle WF , that takes a tree as an input and outputs a 1 if the tree is well-founded and a 0 otherwise.

Theorem 8. (RCA_0) *The following are equivalent:*

- (1) $\Pi_1^1\text{-CA}_0$: *If $\varphi(n)$ is a Π_1^1 formula, then there is a set Z such that for all n , $n \in Z$ if and only if $\varphi(n)$.*
- (2) $\widehat{\text{WF}}$: *If $\langle T_i \rangle_{i \in \mathbb{N}}$ is a sequence of trees, then there is a set Z such that for all n , $n \in Z$ if and only if T has no infinite path.*
- (3) $\widehat{\text{WF}}_L$: *Item (2) for sequences of trees with leaf sets.*
- (4) PK : *Every tree has a perfect kernel.*
- (5) PK_L : *Item (4) for trees with leaf sets.*

Proof. The equivalence of (1) and (2) is Theorem VI.1.1 of Simpson [7]. The equivalence of (1) and (4) is Theorem VI.1.3 of Simpson [7]. The restriction (3) follows trivially from (2), and the converse follows immediately from Theorem 6. By a similar argument, (4) and (5) are equivalent. \square

We now turn to the Weihrauch analogs of the preceding results. The main tool is the computability theoretic version of Lemma 5.

Lemma 9. *There is a uniformly computable map from trees T to T^- , and invertible uniformly computable maps from T to T^+ and T^* . Also, there is a computable functional mapping sequences of trees $\langle T_i \rangle_{i \in \mathbb{N}}$ to $\langle T_i^*, \text{leaf}(T_i^*) \rangle$.*

Proof. The processes described at the beginning of the section are uniformly computable, and for T^+ and T^* , uniformly computably invertible. Leaf sets are uniformly computable for trees of the form T^* . \square

The following Weihrauch analog of Theorem 7 is based on the results of Kihara, Marcone, and Pauly [6].

Theorem 10. $\text{PTT}_1 \equiv_{\text{sW}} \text{PTT}_{1L} <_{\text{W}} \Sigma_1^1\text{-SEP}$. *Also, the following principles are strongly Weihrauch equivalent: $\Sigma_1^1\text{-SEP}$, $\Sigma_1^1\text{-SEP}_L$, $\Sigma_1^1\text{-CA}^-$, and $\Sigma_1^1\text{-CA}_L^-$.*

Proof. The equivalences between the statements and the versions restricted to trees with leaf sets follow from Lemma 9 and Theorem 6. The equivalence of $\Sigma_1^1\text{-SEP}$ and $\Sigma_1^1\text{-CA}^-$ is included in Theorem 3.11 of Kihara, Marcone, and Pauly [6], while $\text{PTT}_1 <_{\text{sW}} \Sigma_1^1\text{-SEP}$ follows from their Corollary 3.7, Theorem 3.11, and Proposition 6.4 [6]. \square

We close the section with the Weihrauch analog of Theorem 8.

Theorem 11. $\text{WF} \equiv_{\text{sW}} \text{WF}_L$. Also, the following four principles are strongly Weihrauch equivalent: $\widehat{\text{WF}}$, $\widehat{\text{WF}}_L$, PK , and PK_L .

Proof. The equivalences $\text{WF} \equiv_{\text{sW}} \text{WF}_L$, $\widehat{\text{WF}} \equiv_{\text{sW}} \widehat{\text{WF}}_L$, and $\text{PK} \equiv_{\text{sW}} \text{PK}_L$ all follow immediately from Lemma 9 and Theorem 6. It suffices to show that $\widehat{\text{WF}} \equiv_{\text{sW}} \text{PK}$.

To see that $\widehat{\text{WF}} \leq_{\text{sW}} \text{PK}$, let $\langle T_i \rangle_{i \in \mathbb{N}}$ be a sequence of trees, the input for $\widehat{\text{WF}}$. For sequences σ and τ with $|\sigma| = |\tau|$, let $\sigma * \tau$ denote the sequence consisting of alternating entries of σ and τ . Thus for σ and τ of length $n+1$, $\sigma * \tau = \langle \sigma(0), \tau(0), \dots, \sigma(n), \tau(n) \rangle$. Define the tree T by including the following sequences for each $i \in \mathbb{N}$:

- $\langle i \rangle \in T$ for each $i \in \mathbb{N}$, and
- if $\sigma \in T_i$, $|\sigma| = n$, and τ is a binary sequence of length n , then $\langle i \rangle \frown (\sigma * \tau) \in T$ and the initial segment of $\langle i \rangle \frown (\sigma * \tau)$ omitting the last element is also in T .

The tree T is uniformly computable from the sequence $\langle T_i \rangle_{i \in \mathbb{N}}$. If T_i has an infinite path p , then for every binary sequence τ , all initial segments of $\langle i \rangle \frown (p * \tau)$ are in T . In this case, there is a perfect subtree of T above $\langle i \rangle$, so $\langle i \rangle$ is in the perfect kernel of T . If T_i is well-founded, then the subtree of extensions of $\langle i \rangle$ in T is also well-founded, so no perfect subtree of T contains $\langle i \rangle$. Thus, if K is a perfect kernel for T , then T_i is well-founded if and only if $\langle i \rangle \in K$. Summarizing, $Z = \{i \mid \langle i \rangle \in K\}$ is the desired output for $\widehat{\text{WF}}$.

To see that $\text{PK} \leq_{\text{sW}} \widehat{\text{WF}}$, let T be an input tree for PK . In the following, we freely conflate finite sequences with their natural number codes. For each finite sequence $\sigma \in T$, define T_σ as follows:

- $\langle \sigma \rangle \in T_\sigma$, and

- if $\langle \sigma, \dots, \langle \tau_0, \dots, \tau_m \rangle \rangle \in T_\sigma$, and for each $i \leq m$, $\tau_i \widehat{e}_{i,0}$ and $\tau_i \widehat{e}_{i,1}$ are incompatible extensions of τ_i in T , then

$$\langle \sigma, \dots, \langle \tau_0, \dots, \tau_m \rangle, \langle \tau_0 \widehat{e}_{0,0}, \tau_0 \widehat{e}_{0,1}, \dots, \tau_m \widehat{e}_{m,0}, \tau_m \widehat{e}_{m,1} \rangle \rangle \in T_\sigma.$$

The sequence $\langle T_\sigma \rangle_{\sigma \in T}$ (which can be viewed as $\langle T_i \rangle_{i \in \mathbb{N}}$) is uniformly computable from T . For each $\sigma \in T$, T_σ has an infinite path if and only if σ is contained in a perfect subtree of T . Let Z be a solution of $\widehat{\text{WF}}$ for $\langle T_\sigma \rangle_{\sigma \in T}$. Then $Z = \{\sigma \in T \mid T_\sigma \text{ is well-founded}\}$, and $K = \{\sigma \in T \mid \sigma \notin Z\}$ is the perfect kernel of T . \square

An application

This section presents a Weihrauch analysis closely related to Theorem 6 of Davis, Hirst, Pardo, and Ransom [4]. A hypergraph $H = (V, E)$ consists of a set of vertices $V = \{v_0, v_1, \dots\}$ and a set of edges $E = \{e_1, e_2, \dots\}$, where each edge in E is a set of vertices. For hypergraphs, an edge can be a set of any cardinality. If every edge of a hypergraph has cardinality exactly 2, then H is a graph. A k -coloring of a hypergraph $H = (V, E)$ is a function $f : V \rightarrow k$. A k -coloring is called proper if every edge with at least two vertices contains vertices of different colors. Let $\text{HPC}(k)$ be the problem that accepts a hypergraph H as input, outputs 1 if H has a proper k -coloring, and outputs 0 otherwise.

The second part of the proof of the following theorem applies a leaf management result from the preceding section.

Theorem 12. *For all $k \geq 2$, $\text{HPC}(k) \equiv_{\text{sW}} \text{WF}$.*

Proof. To see that $\text{HPC}(k) \leq_{\text{sW}} \text{WF}$, let $H = (V, E)$ be a hypergraph input for $\text{HPC}(k)$. In the following, we freely conflate vertices and finite collections of vertices with their integer codes. Build a tree T by including sequences $\sigma = \langle \sigma_0, \sigma_1, \dots, \sigma_m \rangle$ satisfying the following conditions for each $i \leq m$.

- If $i = 2j$, then σ_i is a set of two vertices in edge e_j , or σ_i is a code for \emptyset and e_j does not contain a pair of vertices in the list $\{v_0, \dots, v_m\}$.
- If $i = 2j + 1$, then $\sigma_i < k$. We view this as a color for v_j .
- The partial coloring of H given by the odd entries of σ uses distinct colors on the pairs of vertices listed in the even entries.

The odd entries of any infinite path in T encode a proper k -coloring of H . Also, any proper k -coloring of H can be used to define an infinite path through T . (If H has no edges of cardinality less than 2, an infinite path can be uniformly computed from any proper coloring, but this is not necessary for the current argument.) Thus, $\text{HPC}(k)$ is 1 for H if and only if WF is 0 for T .

By Theorem 11, $\text{WF} \equiv_{\text{sW}} \text{WF}_L$, so to complete the proof it suffices to show that $\text{WF}_L \leq_{\text{sW}} \text{HPC}(k)$. We will prove this for $k = 2$ and indicate how to modify the argument for larger values of k .

Let T be an input for WF_L , that is, a tree with a leaf set. Emulating the construction from the proof of Theorem 6 of Davis et al. [4], define a hypergraph H as follows. The vertices of H include the five vertices $\{a_0, a_1, b_0, b_1, s\}$ plus two vertices labeled σ_0 and σ_1 for each sequence $\sigma \in T$. The edges of H consist of

- (a_0, a_1) , (a_1, s) , (b_0, b_1) , and (b_1, s) ,
- (σ_0, σ_1) for every nonempty $\sigma \in T$,
- (σ_1, s) if σ is a leaf of T ,
- $E_\sigma = \{\sigma_1\} \cup \{\tau_0 \mid \tau \in T \wedge \exists n \tau = \sigma \frown n\}$ if $\sigma \in T$ is not a leaf, and
- $E_0 = \{a_0, b_0\} \cup \{\sigma_0 \mid \sigma \in T \wedge |\sigma| = 1\}$.

H is uniformly computable from T and its leaf set. Note that the leaf set is used in the third and fourth bullets. H has a proper 2-coloring if and only if T has an infinite path. For details, see the proof of Theorem 6 of [4]. Thus $\text{WF}_L \leq_{\text{sW}} \text{HPC}(2)$. To prove the reduction for larger values of k , modify the construction by adding a complete subgraph on $k - 2$ vertices to H and connecting each of its vertices to every vertex of H with an edge consisting of two vertices. \square

Parallelization yields the Weihrauch analog of part of the reverse mathematical Theorem 6 of Davis et al. [4].

Corollary 13. *For $k \geq 2$, $\widehat{\text{WF}} \equiv_{\text{sW}} \widehat{\text{HPC}}(k)$.*

Proof. Fix k . By Theorem 12, $\text{WF} \equiv_{\text{sW}} \text{HPC}(k)$, so by Proposition 3.6 part (3) of Brattka, Gherardi, and Pauly [3], $\widehat{\text{WF}} \equiv_{\text{sW}} \widehat{\text{HPC}}(k)$ \square

The arguments used in Theorem 6 of [4] can be used to extend Theorem 12 and Corollary 13 to conflict-free colorings.

Acknowledgements

A talk related to this paper was presented at Dagstuhl Seminar 18361, organized by Vasco Brattka, Damir Dzhafarov, Alberto Marcone, and Arno Pauly, and held September 2-7 of 2018 at Schloss Dagstuhl, the Leibniz-Zentrum für Informatik [1]. The author’s travel to the seminar was supported by a Board of Trustees travel grant from Appalachian State University.

Bibliography

- [1] Vasco Brattka, Damir Dzhafarov, Alberto Marcone, and Arno Pauly, *Measuring the Complexity of Computational Content: From Combinatorial Problems to Analysis (Dagstuhl Seminar 18361)*, Dagstuhl Reports **8** (2018). To appear.
- [2] 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. MR2760117
- [3] Vasco Brattka, Guido Gherardi, and Arno Pauly, *Weihrauch Complexity in Computable Analysis* (2017), 50+xi pp., available at [arXiv:1707.03202](https://arxiv.org/abs/1707.03202).
- [4] Caleb Davis, Jeffrey Hirst, Jake Pardo, and Tim Ransom, *Reverse mathematics and colorings of hypergraphs*, Archive for Mathematical Logic, posted on November 29, 2018, DOI 10.1007/s00153-018-0654-z.
- [5] Harvey M. Friedman and Jeffrey L. Hirst, *Weak comparability of well orderings and reverse mathematics*, Ann. Pure Appl. Logic **47** (1990), no. 1, 11–29, DOI 10.1016/0168-0072(90)90014-S. MR1050559
- [6] Takayuki Kihara, Alberto Marcone, and Arno Pauly, *Searching for an analogue of ATR_0 in the Weihrauch lattice* (2018), available at [arXiv:1812.01549](https://arxiv.org/abs/1812.01549).
- [7] Stephen G. Simpson, *Subsystems of second order arithmetic*, 2nd ed., Perspectives in Logic, Cambridge University Press, Cambridge; Association for Symbolic Logic, Poughkeepsie, NY, 2009. MR2517689