Decidability and Semidecidability via Ordinals

Nicolai Kraus\textsuperscript{1}, Fredrik Nordvall Forsberg\textsuperscript{2}, and Chuangjie Xu\textsuperscript{3}

\textsuperscript{1} University of Nottingham, Nottingham, UK
\textsuperscript{2} University of Strathclyde, Glasgow, UK
\textsuperscript{3} fortiss GmbH, Munich, Germany

Constructive notions of ordinals. Ordinal numbers, or simply ordinals, can be seen as a generalisation of the natural numbers. They allow the characterisation of possibly infinite orders, and termination proofs are one of their main applications. There are various formulations of (initial segments of) ordinals that coincide in a classical setting, but have very different behaviour constructively. One such formulation is the ordinal notation system of Cantor normal forms (binary trees with a certain property; different presentations are presented in [FXG20]). Another formulation are set-theoretic ordinals (extensional, transitive, wellfounded orders; developed in [Uni13, Chp 10.3] and [Esc21]). Yet another formulation are Brouwer ordinal trees (constructed with zero, successors, and limits; cf. Kleene’s $\mathcal{O}$ [Chu38, Kle38]). We have compared these three formulations at TYPES’21 [KNFX21b], and a full publication is now available [KNFX21a].

Decidability and partial decidability of properties of ordinals. As a rule of thumb, Cantor normal forms have decidable properties as long as we only consider finitely many of them: We can always check whether two given trees are equal, or which of the two represents the larger ordinal, and so on. It is similarly clear that properties of set-theoretic ordinals are generally undecidable: If $P$ is proposition (or subsingleton, i.e. a type with at most one element), then the empty order on $P$ is a set-theoretic ordinal. Being able to decide whether this ordinal $P$ is equal to 1 directly corresponds to the law of excluded middle. Brouwer ordinal trees sit in the “sweet spot” in the middle: some of their properties are fully decidable, but many are “partially” decidable. In this work, we explore how this can be expressed and which statements can be proved internally in homotopy type theory.

Brouwer ordinal trees. Of course, how much is decidable about Brouwer ordinal trees depends on how exactly we define them. A very basic version is the inductive type $\mathcal{O}$ with constructors $\text{zero} : \mathcal{O}$, $\text{suc} : \mathcal{O} \to \mathcal{O}$, and $\text{sup} : (\mathbb{N} \to \mathcal{O}) \to \mathcal{O}$, as sometimes considered in the functional programming community. This type is however too simplistic for our purposes and, unlike the other discussed formulations of ordinals, does not even come with the “correct” notion of equality; if $s : \mathbb{N} \to \mathcal{O}$ is a sequence, then $\text{sup}(s_0, s_1, s_2, \ldots)$ and $\text{sup}(s_1, s_0, s_2, \ldots)$ are different trees. In [KNFX21a], we construct the type $\text{Brw} : \text{hSet}$ mutually with a relation $\leq : \text{Brw} \to \text{Brw} \to \text{hProp}$ quotient inductive-inductively [ACD+18]. The constructors for $\text{Brw}$ are $\text{zero} : \text{Brw}$, $\text{suc} : \text{Brw} \to \text{Brw}$, and $\text{limit} : (\mathbb{N} \twoheadrightarrow \text{Brw}) \to \text{Brw}$, where $\mathbb{N} \twoheadrightarrow \text{Brw}$ are strictly increasing sequences. In addition, we have a path constructor $\text{bisim} : f \approx \leq g \to \text{limit } f = \text{limit } g$, where $f \approx \leq g$ means that $f$ and $g$ are bisimilar sequences.

Decidable properties of Brouwer trees. The notion of decidability is standard: We say that a proposition $P$ is decidable if we have $P + \neg P$, and a predicate $R : \text{Brw} \to \text{hProp}$ is decidable if every $R(x)$ is decidable.

\textsuperscript{*}Supported by the Royal Society, grant reference URF\textbackslash{}R1\textbackslash{}191055, and the UK National Physical Laboratory Measurement Fellowship project Dependent types for trustworthy tools.
Lemma 1. It is decidable whether \( x : \text{Brw} \) is finite, i.e. whether it is in the image of the obvious map \( \mathbb{N} \to \text{Brw} \). If \( n \) is a finite ordinal, then equality with \( n \) and comparisons with \( n \) are decidable; i.e. the predicates \( (\_ = n) \), \( (\_ > n) \), and \( (\_ < n) \) are decidable. Similarly, the predicates \( (\_ \geq \omega) \) and \( (\_ \geq \omega) \) are decidable.

Proof. This is a consequence of the fact that \( \text{Brw} \) satisfies the classification property of \([\text{KNFX}21]\): For a given ordinal, we can check whether it is zero, a successor, or a limit, and a limit cannot be finite. Note that this argument crucially relies on the fact that our limit constructor takes strictly increasing sequences as argument. Without this requirement, all of the properties mentioned in the lemma would be incorrect.

Note that, while \((\_ \geq \omega)\) is decidable, being able to decide \((\_ > \omega)\) is equivalent to LPO, which states: \( \forall f : \mathbb{N} \to 2. (\exists n. f_n = 1) \lor (\forall n. f_n = 0) \).

Partial decidability. While internal decidability is standard, other notions of synthetic computability theory (cf. [BR87, Bau06, FKS19]) are not used as much. In our work, we consider three formulations of partial decidability. Let \( P \) be a proposition:

1. We say that \( P \) is semidecidable if \( \exists \alpha : \mathbb{N} \to \text{Bool}. (\exists i. \alpha_i = 1) \leftrightarrow P \). This definition is due to Bauer [Bau06] (see also [EK17, FLWD+18, dJP22]).
2. Using the free \( \omega \)-complete partial order on the unit type [ADK17], equivalent to the Sierpinski type \( S \) [Vel17, CUV19], one can define \( P \) to be Sierpinski-semidecidable if \( \exists s : S. (s = T) \leftrightarrow P \). This notion of semidecidability has been used by Gilbert [Gil17].
3. If \( x : \text{Brw} \) is an ordinal, we can say that \( P \) can be decided in \( x \) steps if \( \exists y : \text{Brw}. (y > x) \leftrightarrow P \).

Lemma 2. \( P \) is semidecidable if and only if it can be decided in \( \omega \) steps. If \( P \) is semidecidable, then it is also Sierpinski-semidecidable.

Proof. It is easy to translate between semidecidable and decidable in \( \omega \) steps. Given a binary sequence \( \alpha \), we define an increasing sequence \( f : \mathbb{N} \to \text{Brw} \) by \( f_0 \equiv \text{zero} \), \( f_{n+1} \equiv f_n + \omega^{\alpha_n} \).

Similarly, given \( x : \text{Brw} \), we define a binary sequence \( \alpha \) as follows: if \( x \) is finite, then \( \alpha \) is constantly 0. Otherwise, if \( x \) is a successor and infinite, \( \alpha \) is constantly 1. The remaining case is that \( x \) is \( \text{limit}(f) \). If \( f_n \) is finite, we define \( \alpha_n \) to be 0, else 1.

In both cases, we have \((\exists i. \alpha_i = 1) \leftrightarrow (x > \omega)\).

In the same way, given a binary sequence \( \alpha \), we immediately get \( s : S \) by taking the least upper bound of \( \alpha \).

A problem that may separate semidecidability and Sierpinski-semidecidability is the following:

Proposition 3. Given \( x : \text{Brw} \) and \( n, k : \mathbb{N} \), it is Sierpinski-semidecidable whether \( x > \omega \cdot n + k \).

Proof. We define \( s_{\omega \cdot n + k}(x) : S \) by induction on \( x \) and transfinite induction on \( \omega \cdot n + k \). The most interesting case is \( s_{\omega \cdot n + 1}(\text{limit}(f)) \), which we define as the least upper bound of the sequence \( i \mapsto s_{\omega \cdot n}(f_i) \). This uses that the limit of a sequence \( f \) is, for any \( i \), at least \( f_i + \omega \).

We expect, but have not proved, that \( x > \omega \cdot n + k \) cannot be decided in \( \omega \) steps. Finally, we give examples for properties that are decidable in a number of steps different from \( \omega \):

Proposition 4. The following three statements are equivalent: (1) \( P \) is decidable; (2) \( P \) is decidable in 0 steps; (3) \( P \) is decidable in a finite number of steps.

Proposition 5. The twin prime conjecture is decidable in \( \omega^3 \) steps. More generally, if \( P : \mathbb{N} \to \text{hProp} \) is semidecidable and \( P(n + 1) \) implies \( P(n) \), then \( \forall n.P(n) \) is decidable in \( \omega^3 \) steps.

In the future, we hope to explore internal semidecidability and its applications further.
References


