A Certified Library of Ordinal Arithmetic

Nicolai Kraus Fredrik Nordvall Forsberg Chuangjie Xu

Continuity, Computability, Constructivity: From Logic to Algorithms

September 20–24 2021, Birmingham/online

What are ordinals?

One answer: **Numbers** for ranking/ordering

$$0, \quad 1, \quad 2, \quad \ldots, \quad \omega, \quad \omega+1, \quad \ldots, \quad \omega\cdot 2, \quad \omega\cdot 2+1, \quad \ldots, \quad \omega\cdot 3, \quad \ldots$$

$$\omega^2, \quad \ldots, \quad \omega^2\cdot 3+\omega\cdot 7+13, \quad \ldots, \quad \omega^\omega, \quad \ldots, \quad \varepsilon_0 = \omega^{\omega^{\omega^{\cdots}}}, \quad \ldots, \quad \varepsilon_{17}, \quad \ldots$$

Another answer: **Sets with an order** $<$ which is

- **transitive**: $(a < b) \to (b < c) \to (a < c)$
- **wellfounded**: every sequence $a_0 > a_1 > a_2 > a_3 > \ldots$ terminates
- and **trichotomous**: $(a < b) \vee (a = b) \vee (b < a)$
- … or **extensional** (instead of trichotomous):

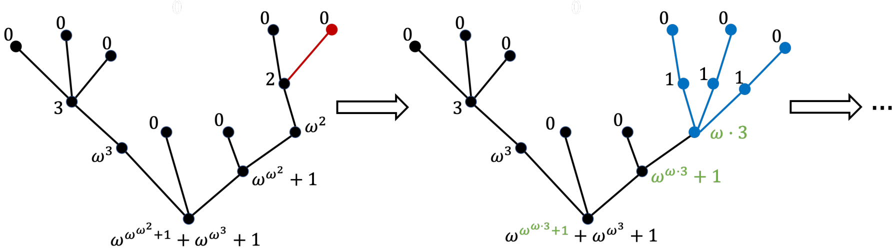
$$(\forall a. a < b \leftrightarrow a < c) \to b = c$$

What are ordinals good for?

Some examples:

- Justifying recursive definitions, e.g., the Ackermann function
- Consistency proof e.g. of Peano's axioms [Gentzen 1936]
- Termination of processes, e.g., [Goodstein 1944], [Turing 1949],

  Hydra game [Kirby&Paris 1982]: All hydras eventually die.



Constructive notions of ordinals

In a constructive setting: different definitions differ!

$$\begin{tikzcd}[column sep=small]
\text{decidable} & \boxed{\text{Cantor normal forms}} \arrow[d, hook, "\text{preserves and reflects } <\text{, commutes with } +, *, \omega^{-}\text{, } \dots"] \\
\text{partially decidable} & \text{Brouwer trees} \arrow[d, hook, "\text{preserves } <\text{, over-approximates } +, *\text{, commutes with limits, } \dots"] \\
\text{undecidable} & \text{Wellfounded \& extensional \& transitive orders}
\end{tikzcd}$$

N. Kraus and F. Nordvall Forsberg and C. Xu. *Connecting constructive notions of ordinals in homotopy type theory*. MFCS 2021.

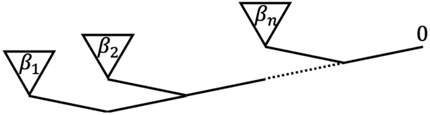
- an axiomatic framework for ordinals and ordinal arithmetic
- connections between the three notions and their arithmetic operations

Cantor normal forms

Motivation: $\alpha = \omega^{\beta_1} + \omega^{\beta_2} + \cdots + \omega^{\beta_n}$ with $\beta_1 \geq \beta_2 \geq \cdots \geq \beta_n$

Let $\mathcal{T}$ be the type of *unlabeled binary trees*:

$$\begin{array}{lcl} 0 & : & \mathcal{T} \\ \omega^{-} + - & : & \mathcal{T} \to \mathcal{T} \to \mathcal{T} \end{array}$$

$\alpha =$ 

Problem: e.g., $\omega^{\beta_1} + \omega^{\beta_2} + 0 \neq \omega^{\beta_2} + \omega^{\beta_1} + 0$

Let $<$ be the *lexicographical order* on binary trees.

A tree is a *Cantor normal form* (Cnf) if $\beta_1 \geq \beta_2 \geq \cdots \geq \beta_n$ and β_i's are Cnfs.

Equivalent implementations: (i) hereditary descending lists (mutually with $<$), and (ii) finite hereditary multisets (as a quotient inductive type)

What Cnf can and cannot do

- The order $<$ on Cnf is extensional, trichotomous and wellfounded.
  - Cnf is an ordinal (in the set-theoretic sense).
  - Cnf satisfies the principle of transfinite induction.
- Every Cnf is a zero, a successor or a limit (of its fundamental sequence).
- Cnf has addition, multiplication and exponentiation (with base ω).
- Cnf cannot calculate limits of sequences.
  - The existence of the limit of ω, ω^{ω}, $\omega^{\omega^{\omega}}$, $\omega^{\omega^{\omega^{\omega}}}$, ... (which is ε_0) implies $\perp$.
  - Cnf represents only the ordinals below ε_0.
- If Cnf has limits of arbitrary *bounded* sequences, then WLPO holds.

Ordinal arithmetic

Theorem. Cnf has addition, multiplication and exponentiation (with base ω).

However, what does it mean?

E.g., we define an operation $+$ on Cnf, why can we call it addition?

Our answer: the set-theoretic (i.e., transfinite-recursive) definition

E.g., we show that our $+$ satisfies

- $a + 0 = a$
- $a + (b + 1) = a + b + 1$
- b is-lim-of $f \to c$ is-lim-of $(\lambda i.a + fi) \to a + b = c$

Note: Cnf cannot calculate limits.

Ordinal arithmetic: addition

Addition is defined inductively on the arguments.

E.g., $$(\omega^a + c) \ + \ (\omega^b + d) = \begin{cases} \omega^b + d & \text{if } a < b \\ \omega^a + \left(c + (\omega^b + d)\right) & \text{otherwise} \end{cases}$$

To verify "b is-lim-of $f \to c$ is-lim-of $(\lambda i.a + fi) \to a + b = c$", we use the following subtraction lemma.

Lemma. If $a \leq b$, then there exists a unique Cnf c such that $a + c = b$.

Similarly, we construct division for verifying the correctness of multiplication (and logarithm for exponentiation).

Commutative Hessenberg sum and product

Consider an equivalent structure that ignores the order:

Quotient inductive type Fhm : hSet of *finite hereditary multisets*, generated by

- 0 : Fhm
- $\omega^{-} \oplus -$: Fhm $\to$ Fhm $\to$ Fhm
- $\mathsf{per}(a, b, c) : \omega^{a} \oplus \omega^{b} \oplus c = \omega^{b} \oplus \omega^{a} \oplus c$ for a, b, c : Fhm

Example: $\omega^{0} \oplus \omega^{\omega^{\omega^{0} \oplus 0} \oplus 0} \oplus \omega^{\omega^{0} \oplus 0} \oplus 0 = \omega^{\omega^{\omega^{0} \oplus 0} \oplus 0} \oplus \omega^{0} \oplus \omega^{\omega^{0} \oplus 0} \oplus 0$ by per, representing the ordinal $\omega^{\omega} + \omega + 1$.

Hessenberg sum is given by the union operation which is commutative.

Commutative product is also conveniently defined.

Equivalence between Cnf and Fhm

- Cnf – descending lists
- Fhm – lists quotient by permutation

$$\begin{tikzcd} \mathsf{Cnf} \arrow[r, bend left, "\text{``identity''}"] & \mathsf{Fhm} \arrow[l, bend left, "\text{insertion sort}"] \end{tikzcd}$$

The constructors $\omega^{-} \oplus -$ and per of Fhm "determine" the insertion on Cnf.

Hessenberg sum and product on Cnf, as well as their commutativity proofs, are obtained by transporting those on Fhm.

Summary

A certified library of a notation system representing ordinals below ε_0:

- transfinite induction
- classification (zero, successor or limit)
- ordinary arithmetic (addition, subtraction, multiplication, division, exponentiation) and correctness proofs
- commutative Hessenberg sum and product
- development in cubical Agda:
  `https://cj-xu.github.io/agda/CertifiedOrdinalArithmetic/`
- connections to Brouwer trees and extensional wellfounded orders

THANK YOU!!