Corecursion and Coinduction

Gabriel Field

15/Aug/2025

Goals

Actual goals:

- Introduce the **conatural numbers**
- Introduce **corecursion** to specifying functions into $\mathrm{co}\mathbb{N}$
- Introduce **coinduction** to reason about corecursive functions
- **Generalise** to other corecursive structures
- Give **useful examples** of coinductive proofs

Notable omissions:

- Exhaustive examples (no time)
- Solid background theory (too much category theory)
- Initial algebras and terminal coalgebras in categories other than $\mathbf{Set}$ (prerequisites not met)

Outline

Outline

The Natural Numbers

Coindustion $\leftarrow$ Corecursion $\leftarrow$ Recursion $\leftarrow \mathbb{N}$.

The Natural Numbers

Coinduction $\leftarrow$ Corecursion $\leftarrow$ Recursion $\leftarrow \mathbb{N}$.

Loose Definition

The set of **natural numbers** is $\mathbb{N} := \{0, 1, \ldots\}$, "freely generated" by the constant 0 and the successor operation $\mathrm{succ} : x \mapsto x + 1$.

The Natural Numbers

Coinduction $\leftarrow$ Corecursion $\leftarrow$ Recursion $\leftarrow \mathbb{N}$.

Loose Definition

The set of **natural numbers** is $\mathbb{N} := \{0, 1, \ldots\}$, "freely generated" by the constant 0 and the successor operation $\mathrm{succ} : x \mapsto x + 1$.

Recursion Principle

Given a set X, a constant $x_0 \in X$, and an operation $f : X \to X$, there is a unique map $u : \mathbb{N} \to X$ with

$$u : 0 \mapsto x_0$$
$$u : \mathrm{succ}(n) \mapsto f(u(n))$$

Recursion from $\mathbb{N}$

Example: Powers

The function $u : \mathbb{N} \to \mathbb{R}$, $u : n \mapsto 3^n$ is defined recursively by

$$u : 0 \mapsto 1 \qquad\qquad u : \mathrm{succ}(n) \mapsto 3^n \cdot n$$

$$3^0 := 1 \qquad\qquad 3^{\mathrm{succ}(n)} := 3^n \cdot n$$

Example: Addition on $\mathbb{N}$

The function $(- + -) : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ is defined recursively by

$$0 + y := y \qquad\qquad \mathrm{succ}(x) + y := \mathrm{succ}(x + y)$$

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\mathrm{lhs}, \mathrm{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\mathrm{lhs}(0) = \mathrm{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \mathrm{lhs}(n) = \mathrm{rhs}(n) \implies \mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{rhs}(\mathrm{succ}(n)))$

then $\mathrm{lhs} = \mathrm{rhs}$.

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\mathrm{lhs}, \mathrm{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\mathrm{lhs}(0) = \mathrm{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \mathrm{lhs}(n) = \mathrm{rhs}(n) \implies \mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{rhs}(\mathrm{succ}(n)))$

then $\mathrm{lhs} = \mathrm{rhs}$.

$$0 + y := y \qquad\qquad \mathrm{succ}(x) + y := \mathrm{succ}(x + y)$$

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\mathrm{lhs}, \mathrm{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\mathrm{lhs}(0) = \mathrm{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \mathrm{lhs}(n) = \mathrm{rhs}(n) \implies \mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{rhs}(\mathrm{succ}(n)))$

then $\mathrm{lhs} = \mathrm{rhs}$.

$$0 + y := y \qquad \mathrm{succ}(x) + y := \mathrm{succ}(x + y)$$

Example: $\forall n \in \mathbb{N}, n + 0 = n$

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\mathrm{lhs}, \mathrm{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\mathrm{lhs}(0) = \mathrm{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \mathrm{lhs}(n) = \mathrm{rhs}(n) \implies \mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{rhs}(\mathrm{succ}(n)))$

then $\mathrm{lhs} = \mathrm{rhs}$.

$$0 + y := y \qquad\qquad \mathrm{succ}(x) + y := \mathrm{succ}(x + y)$$

Example: $\forall n \in \mathbb{N}, n + 0 = n$

Let $\mathrm{lhs} : n \mapsto n + 0$ and $\mathrm{rhs} : n \mapsto n$. Then,

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\text{lhs}, \text{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\text{lhs}(0) = \text{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \text{lhs}(n) = \text{rhs}(n) \implies \text{lhs}(\text{succ}(n)) = \text{rhs}(\text{succ}(n)))$

then $\text{lhs} = \text{rhs}$.

$$0 + y := y \qquad \text{succ}(x) + y := \text{succ}(x + y)$$

Example: $\forall n \in \mathbb{N}, n + 0 = n$

Let $\text{lhs} : n \mapsto n + 0$ and $\text{rhs} : n \mapsto n$. Then,

- $\text{lhs}(0) = 0 + 0 = 0 = \text{rhs}(0)$;

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\mathrm{lhs}, \mathrm{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\mathrm{lhs}(0) = \mathrm{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \mathrm{lhs}(n) = \mathrm{rhs}(n) \implies \mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{rhs}(\mathrm{succ}(n)))$

then $\mathrm{lhs} = \mathrm{rhs}$.

$$0 + y := y \qquad\qquad \mathrm{succ}(x) + y := \mathrm{succ}(x + y)$$

Example: $\forall n \in \mathbb{N}, n + 0 = n$

Let $\mathrm{lhs} : n \mapsto n + 0$ and $\mathrm{rhs} : n \mapsto n$. Then,

- $\mathrm{lhs}(0) = 0 + 0 = 0 = \mathrm{rhs}(0)$;
- $\mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{succ}(n) + 0 = \mathrm{succ}(n + 0) = \mathrm{succ}(n) = \mathrm{rhs}(\mathrm{succ}(n))$;

Induction from $\mathbb{N}$

Theorem: Simple induction

Let $\mathrm{lhs}, \mathrm{rhs} : \mathbb{N} \to X$ be two functions out of $\mathbb{N}$. If

- $\mathrm{lhs}(0) = \mathrm{rhs}(0)$, and
- $(\forall n \in \mathbb{N}, \mathrm{lhs}(n) = \mathrm{rhs}(n) \implies \mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{rhs}(\mathrm{succ}(n)))$

then $\mathrm{lhs} = \mathrm{rhs}$.

$$0 + y := y \qquad \mathrm{succ}(x) + y := \mathrm{succ}(x + y)$$

Example: $\forall n \in \mathbb{N}, n + 0 = n$

Let $\mathrm{lhs} : n \mapsto n + 0$ and $\mathrm{rhs} : n \mapsto n$. Then,

- $\mathrm{lhs}(0) = 0 + 0 = 0 = \mathrm{rhs}(0)$;
- $\mathrm{lhs}(\mathrm{succ}(n)) = \mathrm{succ}(n) + 0 = \mathrm{succ}(n + 0) = \mathrm{succ}(n) = \mathrm{rhs}(\mathrm{succ}(n))$;

so $\mathrm{lhs} = \mathrm{rhs}$.

Outline

The Conatural Numbers

- $\mathbb{N} \to$ Recursion $\to$ Induction
- $\mathrm{co}\mathbb{N} \to$ Corecursion $\to$ Coinduction

The Conatural Numbers

- $\mathbb{N} \to$ Recursion $\to$ Induction
- $\mathrm{co}\mathbb{N} \to$ Corecursion $\to$ Coinduction

Definition: Conatural numbers, Predecessor

The set of **conatural numbers** is $\mathrm{co}\mathbb{N} := \mathbb{N} \sqcup \{\infty\}$.
The **predecessor** operation is the bijection

$$\begin{aligned} \mathrm{pred} : \mathrm{co}\mathbb{N} &\longrightarrow \{\mathrm{no}\} \sqcup \mathrm{co}\mathbb{N} \\ 0 &\longmapsto \mathrm{no} \\ \mathrm{succ}(n) &\longmapsto n \\ \infty &\longmapsto \infty \end{aligned}$$

The Conatural Numbers

- $\mathbb{N} \to$ Recursion $\to$ Induction
- $\mathrm{co}\mathbb{N} \to$ Corecursion $\to$ Coinduction

Definition: Conatural numbers, Predecessor

The set of **conatural numbers** is $\mathrm{co}\mathbb{N} := \mathbb{N} \sqcup \{\infty\}$.
The **predecessor** operation is the bijection

$$\begin{aligned} \mathrm{pred} : \mathrm{co}\mathbb{N} &\longrightarrow \{\mathrm{no}\} \sqcup \mathrm{co}\mathbb{N} \\ 0 &\longmapsto \mathrm{no} \\ \mathrm{succ}(n) &\longmapsto n \\ \infty &\longmapsto \infty \end{aligned}$$

$\mathbb{N}$ comes with $\{\mathrm{no}\} \sqcup \mathbb{N} \xrightarrow{(0,\mathrm{succ})} \mathbb{N}$.

$\mathrm{co}\mathbb{N}$ comes with $\mathrm{co}\mathbb{N} \xrightarrow{\mathrm{pred}} \{\mathrm{no}\} \sqcup \mathbb{N}$.

Partial Functions

Definition: Partial function

A **partial function** $X \nrightarrow Y$ is a function $X \to \{\mathrm{no}\} \sqcup Y$.

Examples:

Partial Functions

Definition: Partial function

A **partial function** $X \nrightarrow Y$ is a function $X \to \{\mathrm{no}\} \sqcup Y$.

Examples:

- $f : x \mapsto 1/x : \mathbb{R} \nrightarrow \mathbb{R}$;
  $f(0) = \mathrm{no}.$

Partial Functions

Definition: Partial function

A **partial function** $X \nrightarrow Y$ is a function $X \to \{\mathrm{no}\} \sqcup Y$.

Examples:

- $f : x \mapsto 1/x : \mathbb{R} \nrightarrow \mathbb{R}$;
  $f(0) = \mathrm{no}$.

- "Get input from user" : `ComputerState` $\nrightarrow$ `String`;
  Fails when the user doesn't give input.

Partial Functions

Definition: Partial function

A **partial function** $X \nrightarrow Y$ is a function $X \to \{\mathrm{no}\} \sqcup Y$.

Examples:

- $f : x \mapsto 1/x : \mathbb{R} \nrightarrow \mathbb{R}$;
  $f(0) = \mathrm{no}$.
- "Get input from user" : `ComputerState` $\nrightarrow$ `String`;
  Fails when the user doesn't give input.
- $\mathrm{pred} : \mathrm{co}\mathbb{N} \nrightarrow \mathrm{co}\mathbb{N}$;
  $\mathrm{pred}(0)$ fails.

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \nrightarrow X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \nrightarrow X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

- $u(x) = 0 \iff f(x) = \mathrm{no}$

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \nrightarrow X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

- $u(x) = 0 \iff f(x) = \mathrm{no}$
- $u(x) = 1 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) = \mathrm{no}$

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \nrightarrow X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

- $u(x) = 0 \iff f(x) = \mathrm{no}$
- $u(x) = 1 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) = \mathrm{no}$
- $u(x) = 2 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) \neq \mathrm{no}$ and $f^3(x) = \mathrm{no}$
- etc.

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \nrightarrow X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

- $u(x) = 0 \iff f(x) = \mathrm{no}$
- $u(x) = 1 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) = \mathrm{no}$
- $u(x) = 2 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) \neq \mathrm{no}$ and $f^3(x) = \mathrm{no}$
- etc.
- $u(x) = \infty \iff f(x) \neq \mathrm{no}$ and $f^2(x) \neq \mathrm{no}$ and $\cdots$

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \nrightarrow X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

- $u(x) = 0 \iff f(x) = \mathrm{no}$
- $u(x) = 1 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) = \mathrm{no}$
- $u(x) = 2 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) \neq \mathrm{no}$ and $f^3(x) = \mathrm{no}$
- etc.
- $u(x) = \infty \iff f(x) \neq \mathrm{no}$ and $f^2(x) \neq \mathrm{no}$ and $\cdots$

Write $u := \mathrm{wait}(f)$.

Corecursion into $\mathrm{co}\mathbb{N}$

Corecursion principle

Let $f : X \rightharpoonup X$ be a partial function. Construct $u : X \to \mathrm{co}\mathbb{N}$ by

- $u(x) = 0 \iff f(x) = \mathrm{no}$
- $u(x) = 1 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) = \mathrm{no}$
- $u(x) = 2 \iff f(x) \neq \mathrm{no}$ and $f(f(x)) \neq \mathrm{no}$ and $f^3(x) = \mathrm{no}$
- etc.
- $u(x) = \infty \iff f(x) \neq \mathrm{no}$ and $f^2(x) \neq \mathrm{no}$ and $\cdots$

Write $u := \mathrm{wait}(f)$.

Lemma: Pred-wait

Let $f : X \rightharpoonup X$ be a partial function and $x \in X$.

- If $f(x) = \mathrm{no}$, then $\mathrm{pred}(\mathrm{wait}(f)(x)) = \mathrm{no}$.
- If $f(x) = x' \neq \mathrm{no}$, then $\mathrm{pred}(\mathrm{wait}(f)(x)) = \mathrm{wait}(f)(x')$.

Abstract nonsense: pred is the terminal partial endo-function.

Corecursion into $\mathrm{co}\mathbb{N}$

Example:

- Let $c : \mathbb{Z}_{>0} \nrightarrow \mathbb{Z}_{>0}$ by

$$c(1) := \mathrm{no}; \quad c(n) := n/2 \text{ if } n \text{ is even}; \quad c(n) := 3n + 1 \text{ otherwise}$$

Corecursion into $\mathrm{co}\mathbb{N}$

Example:

- Let $c : \mathbb{Z}_{>0} \nrightarrow \mathbb{Z}_{>0}$ by

  $c(1) := \mathrm{no}; \quad c(n) := n/2$ if n is even; $\quad c(n) := 3n + 1$ otherwise

- $\mathrm{wait}(c)(x)$ is the length of $x \overset{c}{\mapsto} \cdots \overset{c}{\mapsto} 1$, possibly ∞.

Corecursion into $\mathrm{co}\mathbb{N}$

Example:

- Let $c : \mathbb{Z}_{>0} \nrightarrow \mathbb{Z}_{>0}$ by

  $c(1) := \mathrm{no}; \quad c(n) := n/2$ if n is even; $\quad c(n) := 3n + 1$ otherwise

- $\mathrm{wait}(c)(x)$ is the length of $x \overset{c}{\mapsto} \cdots \overset{c}{\mapsto} 1$, possibly ∞.

- E.g. $5 \overset{c}{\mapsto} 16 \overset{c}{\mapsto} 8 \overset{c}{\mapsto} 4 \overset{c}{\mapsto} 2 \overset{c}{\mapsto} 1 \overset{c}{\mapsto} \mathrm{no}$, so $\mathrm{wait}(c)(5) = 5$.

Corecursion into $\mathrm{co}\mathbb{N}$

Example:

- Let $c : \mathbb{Z}_{>0} \nrightarrow \mathbb{Z}_{>0}$ by

  $c(1) := \mathrm{no}; \quad c(n) := n/2$ if n is even; $\quad c(n) := 3n + 1$ otherwise

- $\mathrm{wait}(c)(x)$ is the length of $x \overset{c}{\mapsto} \cdots \overset{c}{\mapsto} 1$, possibly ∞.

- E.g. $5 \overset{c}{\mapsto} 16 \overset{c}{\mapsto} 8 \overset{c}{\mapsto} 4 \overset{c}{\mapsto} 2 \overset{c}{\mapsto} 1 \overset{c}{\mapsto} \mathrm{no}$, so $\mathrm{wait}(c)(5) = 5$.

- The Collatz conjecture asks whether $\mathrm{wait}(c)(n) \neq \infty$ for all $n \in \mathbb{Z}_{>0}$.

Corecursion into $\mathrm{co}\mathbb{N}$

Definition: Addition in $\mathrm{co}\mathbb{N}$

The function $(- + -) : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}$ is defined by

$$(- + -) := \mathrm{wait}(f)$$

where $f : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \nrightarrow \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N}$ is given by

$$f : (x, y) \mapsto \begin{cases} \mathrm{no} & \text{if } \mathrm{pred}(x) = \mathrm{pred}(y) = \mathrm{no} \\ (x, y') & \text{if } \mathrm{pred}(x) = \mathrm{no} \text{ and } \mathrm{pred}(y) = y' \neq \mathrm{no} \\ (x', y) & \text{if } \mathrm{pred}(x) = x' \neq \mathrm{no} \end{cases}$$

Corecursion into $\mathrm{co}\mathbb{N}$

Definition: Addition in $\mathrm{co}\mathbb{N}$

The function $(- + -) : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}$ is defined by

$$(- + -) := \mathrm{wait}(f)$$

where $f : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \nrightarrow \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N}$ is given by

$$f : (x, y) \mapsto \begin{cases} \mathrm{no} & \text{if } \mathrm{pred}(x) = \mathrm{pred}(y) = \mathrm{no} \\ (x, y') & \text{if } \mathrm{pred}(x) = \mathrm{no} \text{ and } \mathrm{pred}(y) = y' \neq \mathrm{no} \\ (x', y) & \text{if } \mathrm{pred}(x) = x' \neq \mathrm{no} \end{cases}$$

By the pred-wait lemma,

Corecursion into $\mathrm{co}\mathbb{N}$

Definition: Addition in $\mathrm{co}\mathbb{N}$

The function $(- + -) : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}$ is defined by

$$(- + -) := \mathrm{wait}(f)$$

where $f : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \nrightarrow \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N}$ is given by

$$f : (x, y) \mapsto \begin{cases} \mathrm{no} & \text{if } \mathrm{pred}(x) = \mathrm{pred}(y) = \mathrm{no} \\ (x, y') & \text{if } \mathrm{pred}(x) = \mathrm{no} \text{ and } \mathrm{pred}(y) = y' \neq \mathrm{no} \\ (x', y) & \text{if } \mathrm{pred}(x) = x' \neq \mathrm{no} \end{cases}$$

By the pred-wait lemma,

- $\mathrm{pred}(0 + 0) = \mathrm{no}$, so $0 + 0 = 0$;

Corecursion into $\mathrm{co}\mathbb{N}$

Definition: Addition in $\mathrm{co}\mathbb{N}$

The function $(- + -) : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}$ is defined by

$$(- + -) := \mathrm{wait}(f)$$

where $f : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \nrightarrow \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N}$ is given by

$$f : (x, y) \mapsto \begin{cases} \mathrm{no} & \text{if } \mathrm{pred}(x) = \mathrm{pred}(y) = \mathrm{no} \\ (x, y') & \text{if } \mathrm{pred}(x) = \mathrm{no} \text{ and } \mathrm{pred}(y) = y' \neq \mathrm{no} \\ (x', y) & \text{if } \mathrm{pred}(x) = x' \neq \mathrm{no} \end{cases}$$

By the pred-wait lemma,

- $\mathrm{pred}(0 + 0) = \mathrm{no}$, so $0 + 0 = 0$;
- $\mathrm{pred}(0 + y) = \mathrm{pred}(y)$, so $0 + y = y$;

Corecursion into $\mathrm{co}\mathbb{N}$

Definition: Addition in $\mathrm{co}\mathbb{N}$

The function $(- + -) : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}$ is defined by

$$(- + -) := \mathrm{wait}(f)$$

where $f : \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N} \nrightarrow \mathrm{co}\mathbb{N} \times \mathrm{co}\mathbb{N}$ is given by

$$f : (x, y) \mapsto \begin{cases} \mathrm{no} & \text{if } \mathrm{pred}(x) = \mathrm{pred}(y) = \mathrm{no} \\ (x, y') & \text{if } \mathrm{pred}(x) = \mathrm{no} \text{ and } \mathrm{pred}(y) = y' \neq \mathrm{no} \\ (x', y) & \text{if } \mathrm{pred}(x) = x' \neq \mathrm{no} \end{cases}$$

By the pred-wait lemma,

- $\mathrm{pred}(0 + 0) = \mathrm{no}$, so $0 + 0 = 0$;
- $\mathrm{pred}(0 + y) = \mathrm{pred}(y)$, so $0 + y = y$;
- $\mathrm{pred}(x + y) = \mathrm{pred}(x) + y$ whenever $\mathrm{pred}(x) \neq \mathrm{no}$.

Outline

Coinduction into $\mathrm{co}\mathbb{N}$

Given $f, g : X \to Y$, let $Y_{/f=g}$ be the quotient of Y by the smallest equivalence relation $\sim$ with $f(x) \sim g(x)$ for each $x \in X$.

Coinduction into $\mathrm{co}\mathbb{N}$

Given $f, g : X \to Y$, let $Y_{/f=g}$ be the quotient of Y by the smallest equivalence relation $\sim$ with $f(x) \sim g(x)$ for each $x \in X$.

So, $[f(x)] = [g(x)]$ in $Y_{/f=g}$.

Coinduction into $\mathrm{co}\mathbb{N}$

Given $f, g : X \to Y$, let $Y_{/f=g}$ be the quotient of Y by the smallest equivalence relation $\sim$ with $f(x) \sim g(x)$ for each $x \in X$.

So, $[f(x)] = [g(x)]$ in $Y_{/f=g}$.

Theorem: Simple coinduction

Let $\mathrm{lhs}, \mathrm{rhs} : X \to \mathrm{co}\mathbb{N}$ be two functions into $\mathrm{co}\mathbb{N}$.
Let $[-] : \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$ be the quotient map.
If

$$\forall x \in X, \quad [\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$$

(with $[\mathrm{no}] := \mathrm{no}$)

Coinduction into $\mathrm{co}\mathbb{N}$

Given $f, g : X \to Y$, let $Y_{/f=g}$ be the quotient of Y by the smallest equivalence relation $\sim$ with $f(x) \sim g(x)$ for each $x \in X$.

So, $[f(x)] = [g(x)]$ in $Y_{/f=g}$.

Theorem: Simple coinduction

Let $\mathrm{lhs}, \mathrm{rhs} : X \to \mathrm{co}\mathbb{N}$ be two functions into $\mathrm{co}\mathbb{N}$.
Let $[-] : \mathrm{co}\mathbb{N} \to \mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$ be the quotient map.
If

$$\forall x \in X, \quad [\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$$

(with $[\mathrm{no}] := \mathrm{no}$), then $\mathrm{lhs} = \mathrm{rhs}$.

Coinduction into $\mathrm{co}\mathbb{N}$: proof sketch

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof sketch.

It follows that $[\mathrm{pred}^n(\mathrm{lhs}(x))] = [\mathrm{pred}^n(\mathrm{rhs}(x))]$ for all n (with $\mathrm{pred}(\mathrm{no}) := \mathrm{no}$).

Coinduction into $\mathrm{co}\mathbb{N}$: proof sketch

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof sketch.
It follows that $[\mathrm{pred}^n(\mathrm{lhs}(x))] = [\mathrm{pred}^n(\mathrm{rhs}(x))]$ for all n (with $\mathrm{pred}(\mathrm{no}) := \mathrm{no}$).

- If $\mathrm{lhs}(x) = n \in \mathbb{N}$, then

$$[\mathrm{pred}^{n+1}(\mathrm{rhs}(x))] = [\mathrm{pred}^{n+1}(\mathrm{lhs}(x))] = [\mathrm{no}] = \mathrm{no}$$

  so $\mathrm{pred}^{n+1}(\mathrm{rhs}(x)) = \mathrm{no}$, and hence $\mathrm{rhs}(x) = n = \mathrm{lhs}(x)$.

Coinduction into $\mathrm{co}\mathbb{N}$: proof sketch

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof sketch.

It follows that $[\mathrm{pred}^n(\mathrm{lhs}(x))] = [\mathrm{pred}^n(\mathrm{rhs}(x))]$ for all n (with $\mathrm{pred}(\mathrm{no}) := \mathrm{no}$).

- If $\mathrm{lhs}(x) = n \in \mathbb{N}$, then

$$[\mathrm{pred}^{n+1}(\mathrm{rhs}(x))] = [\mathrm{pred}^{n+1}(\mathrm{lhs}(x))] = [\mathrm{no}] = \mathrm{no}$$

  so $\mathrm{pred}^{n+1}(\mathrm{rhs}(x)) = \mathrm{no}$, and hence $\mathrm{rhs}(x) = n = \mathrm{lhs}(x)$.
- If $\mathrm{rhs}(x) \in \mathbb{N}$, do the same thing.

Coinduction into $\mathrm{co}\mathbb{N}$: proof sketch

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof sketch.

It follows that $[\mathrm{pred}^n(\mathrm{lhs}(x))] = [\mathrm{pred}^n(\mathrm{rhs}(x))]$ for all n (with $\mathrm{pred}(\mathrm{no}) := \mathrm{no}$).

- If $\mathrm{lhs}(x) = n \in \mathbb{N}$, then

$$[\mathrm{pred}^{n+1}(\mathrm{rhs}(x))] = [\mathrm{pred}^{n+1}(\mathrm{lhs}(x))] = [\mathrm{no}] = \mathrm{no}$$

  so $\mathrm{pred}^{n+1}(\mathrm{rhs}(x)) = \mathrm{no}$, and hence $\mathrm{rhs}(x) = n = \mathrm{lhs}(x)$.
- If $\mathrm{rhs}(x) \in \mathbb{N}$, do the same thing.
- The only remaining case is $\mathrm{lhs}(x) = \infty = \mathrm{rhs}(x)$.

Coinduction into $\mathrm{co}\mathbb{N}$: proof sketch

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof sketch.

It follows that $[\mathrm{pred}^n(\mathrm{lhs}(x))] = [\mathrm{pred}^n(\mathrm{rhs}(x))]$ for all n (with $\mathrm{pred}(\mathrm{no}) := \mathrm{no}$).

- If $\mathrm{lhs}(x) = n \in \mathbb{N}$, then

$$[\mathrm{pred}^{n+1}(\mathrm{rhs}(x))] = [\mathrm{pred}^{n+1}(\mathrm{lhs}(x))] = [\mathrm{no}] = \mathrm{no}$$

  so $\mathrm{pred}^{n+1}(\mathrm{rhs}(x)) = \mathrm{no}$, and hence $\mathrm{rhs}(x) = n = \mathrm{lhs}(x)$.
- If $\mathrm{rhs}(x) \in \mathbb{N}$, do the same thing.
- The only remaining case is $\mathrm{lhs}(x) = \infty = \mathrm{rhs}(x)$.

Therefore, $\mathrm{lhs} = \mathrm{rhs}$.

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

$(*)$ $\mathrm{pred}(x + 0) = \mathrm{no}$ whenever $\mathrm{pred}(x) = \mathrm{no}$

$(**)$ $\mathrm{pred}(x + y) = \mathrm{pred}(x) + y$ whenever $\mathrm{pred}(x) \neq \mathrm{no}$

Proof that $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$.

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

$(*)$ $\mathrm{pred}(x+0) = \mathrm{no}$ whenever $\mathrm{pred}(x) = \mathrm{no}$

Proof that $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$.
Set $\mathrm{lhs} : x \mapsto x + 0$ and $\mathrm{rhs} : x \mapsto x$.

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

$(*)$ $\mathrm{pred}(x+0) = \mathrm{no}$ whenever $\mathrm{pred}(x) = \mathrm{no}$

Proof that $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$**.**
Set $\mathrm{lhs} : x \mapsto x + 0$ and $\mathrm{rhs} : x \mapsto x$.

- If $\mathrm{pred}(x) = \mathrm{no}$, then

$$\begin{aligned}
[\mathrm{pred}(\mathrm{lhs}(x))] &= [\mathrm{pred}(x+0)] \\
&= [\mathrm{no}] \\
&= [\mathrm{pred}(x)] \\
[\mathrm{pred}(\mathrm{lhs}(x))] &= [\mathrm{pred}(\mathrm{rhs}(x))]
\end{aligned}$$

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

$(**)$ $\mathrm{pred}(x+y) = \mathrm{pred}(x) + y$ whenever $\mathrm{pred}(x) \neq \mathrm{no}$

Proof that $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$.
Set $\mathrm{lhs} : x \mapsto x + 0$ and $\mathrm{rhs} : x \mapsto x$.

- If instead $\mathrm{pred}(x) \neq \mathrm{no}$, then

$$\begin{aligned}
[\mathrm{pred}(\mathrm{lhs}(x))] &= [\mathrm{pred}(x+0)] \\
&= [\mathrm{pred}(x) + 0] \\
&= [\mathrm{lhs}(\mathrm{pred}(x))] \\
&= [\mathrm{rhs}(\mathrm{pred}(x))] \\
&= [\mathrm{pred}(x)] \\
[\mathrm{pred}(\mathrm{lhs}(x))] &= [\mathrm{pred}(\mathrm{rhs}(x))]
\end{aligned}$$

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof that $\forall x \in \mathrm{co}\mathbb{N},\ x + 0 = x$.

Set $\mathrm{lhs} : x \mapsto x + 0$ and $\mathrm{rhs} : x \mapsto x$.

- The theorem asserts $\mathrm{lhs} = \mathrm{rhs}$.

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

- $\mathrm{pred}(a + b) = \mathrm{pred}(b)$ whenever $\mathrm{pred}(a) = \mathrm{no}$
- $\mathrm{pred}(a + b) = \mathrm{pred}(a)$ whenever $\mathrm{pred}(b) = \mathrm{no}$
- $\mathrm{pred}(a + b) = \mathrm{pred}(a) + b$ whenever $\mathrm{pred}(a) \neq \mathrm{no}$
- $\mathrm{pred}(a + b) = a + \mathrm{pred}(b)$ whenever $\mathrm{pred}(b) \neq \mathrm{no}$ (possible)

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$.

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$.

- Set $X := \mathrm{co}\mathbb{N}^2$.
- Set $\mathrm{lhs}(x, y) := x + y$.
- Set $\mathrm{rhs}(x, y) := y + x$.

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

- $\mathrm{pred}(a + b) = \mathrm{pred}(b)$ whenever $\mathrm{pred}(a) = \mathrm{no}$
- $\mathrm{pred}(a + b) = \mathrm{pred}(a)$ whenever $\mathrm{pred}(b) = \mathrm{no}$

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$.

- If $\mathrm{pred}(x) = \mathrm{no}$, then

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$

Theorem: Simple coinduction

If $\forall x \in X,\ [\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

- $\mathrm{pred}(a + b) = \mathrm{pred}(b)$ whenever $\mathrm{pred}(a) = \mathrm{no}$
- $\mathrm{pred}(a + b) = \mathrm{pred}(a)$ whenever $\mathrm{pred}(b) = \mathrm{no}$

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$.

- If $\mathrm{pred}(x) = \mathrm{no}$, then

$$\begin{aligned}
[\mathrm{pred}(\mathrm{lhs}(x, y))] &= [\mathrm{pred}(x + y)] \\
&= [\mathrm{pred}(y)] \\
&= [\mathrm{pred}(y + x)] \\
[\mathrm{pred}(\mathrm{lhs}(x, y))] &= [\mathrm{pred}(\mathrm{rhs}(x, y))]
\end{aligned}$$

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

- $\mathrm{pred}(a + b) = \mathrm{pred}(a) + b$ whenever $\mathrm{pred}(a) \neq \mathrm{no}$
- $\mathrm{pred}(a + b) = a + \mathrm{pred}(b)$ whenever $\mathrm{pred}(b) \neq \mathrm{no}$

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$**.**

- If $\mathrm{pred}(x) \neq \mathrm{no}$, then

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

- $\mathrm{pred}(a + b) = \mathrm{pred}(a) + b$ whenever $\mathrm{pred}(a) \neq \mathrm{no}$
- $\mathrm{pred}(a + b) = a + \mathrm{pred}(b)$ whenever $\mathrm{pred}(b) \neq \mathrm{no}$

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$.

- If $\mathrm{pred}(x) \neq \mathrm{no}$, then

$$\begin{aligned}
[\mathrm{pred}(\mathrm{lhs}(x, y))] &= [\mathrm{pred}(x + y)] \\
&= [\mathrm{pred}(x) + y] \\
&= [\mathrm{lhs}(\mathrm{pred}(x), y)] \\
&= [\mathrm{rhs}(\mathrm{pred}(x), y)] \\
&= [y + \mathrm{pred}(x)] \\
&= [\mathrm{pred}(y + x)] = [\mathrm{pred}(\mathrm{rhs}(x, y))]
\end{aligned}$$

Coinduction into $\mathrm{co}\mathbb{N}$: $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$

Theorem: Simple coinduction

If $\forall x \in X$, $[\mathrm{pred}(\mathrm{lhs}(x))] = [\mathrm{pred}(\mathrm{rhs}(x))]$ in $\mathrm{co}\mathbb{N}_{/\mathrm{lhs}=\mathrm{rhs}}$, then $\mathrm{lhs} = \mathrm{rhs}$.

Proof that $\forall x, y \in \mathrm{co}\mathbb{N},\ x + y = y + x$**.**

- The theorem asserts $\mathrm{lhs} = \mathrm{rhs}$.

Other properties of $\mathrm{co}\mathbb{N}$

These can be proven coinductively:

- $\mathrm{pred}(x + y) = x + \mathrm{pred}(y)$ whenever $\mathrm{pred}(y) \neq \mathrm{no}$.
- $+$ is associative.
- If x, y are (finite or infinite) sequences, then $\mathrm{length}(x +\!\!+\, y) = \mathrm{length}(x) + \mathrm{length}(y)$.

Other properties of $\mathrm{co}\mathbb{N}$

Coinductive proofs only involve ≤ 2 cases:

- $\mathrm{pred}(\textsf{something}) = \mathrm{no}$
- $\mathrm{pred}(\textsf{something}) \neq \mathrm{no}$

and sometimes no splits at all!

Other properties of $\mathrm{co}\mathbb{N}$

Coinductive proofs only involve ≤ 2 cases:

- $\mathrm{pred}(\text{something}) = \mathrm{no}$
- $\mathrm{pred}(\text{something}) \neq \mathrm{no}$

and sometimes no splits at all!

Contrast against inductive-plus-infinity proofs with 3 cases:

- $(\text{something}) = 0$
- $(\text{something}) = \mathrm{succ}(\text{something else})$
- $(\text{something}) = \infty$

Why induct when you could coinduct?

Outline

Lists

Let A be a set.

Definition: Lists on A

A **list** on A is a finite-length sequence of elements in A.

Lists

Let A be a set.

Definition: Lists on A

A **list** on A is a finite-length sequence of elements in A.

$$\mathrm{List}(A) := \{[], [a_0], [a_0, a_1], \ldots \mid a_0, a_1, \ldots \in A\}$$

Lists

Let A be a set.

Definition: Lists on A

A **list** on A is a finite-length sequence of elements in A.

$$\mathrm{List}(A) := \{[], [a_0], [a_0, a_1], \ldots \mid a_0, a_1, \ldots \in A\}$$

$\mathrm{List}(A)$ comes with a constant $[] \in \mathrm{List}(A)$, and an operation

$$\begin{aligned} \mathrm{cons} : A \times \mathrm{List}(A) &\longrightarrow \mathrm{List}(A) \\ (a_0, [a_1, \ldots]) &\longmapsto [a_0, a_1, \ldots] \end{aligned}$$

Colists

Let A be a set.

Definition: Colists on A

A **colist** on A is a finite-or-infinite sequence of elements in A.

Colists

Let A be a set.

Definition: Colists on A

A **colist** on A is a finite-or-infinite sequence of elements in A.

$$\mathrm{coList}(A) := \mathrm{List}(A) \sqcup \{[a_0, a_1, \ldots] \mid a_0, a_1, \ldots \in A\}$$

Colists

Let A be a set.

Definition: Colists on A

A **colist** on A is a finite-or-infinite sequence of elements in A.

$$\mathrm{coList}(A) := \mathrm{List}(A) \sqcup \{[a_0, a_1, \ldots] \mid a_0, a_1, \ldots \in A\}$$

$\mathrm{coList}(A)$ comes with an operation

$$\begin{aligned} \mathrm{pop} : \mathrm{coList}(A) &\longrightarrow \{\mathrm{no}\} \sqcup (A \times \mathrm{coList}(A)) \\ [] &\longmapsto \mathrm{no} \\ [a_0, a_1, \ldots] &\longmapsto (a_0, [a_1, \ldots]) \end{aligned}$$

Recursion/Corecursion

Recursion principle

To define a function $u : \mathrm{List}(A) \to X$, it is enough to specify $u([])$ and to specify $u(\mathrm{cons}(a, \mathrm{as}))$ in terms of a and $u(\mathrm{as})$.

Recursion/Corecursion

Recursion principle

To define a function $u : \mathrm{List}(A) \to X$, it is enough to specify $u([])$ and to specify $u(\mathrm{cons}(a, \mathrm{as}))$ in terms of a and $u(\mathrm{as})$.

Corecursion principle

To define a function $u : X \to \mathrm{coList}(A)$, it is enough to specify a partial function $f : X \nrightarrow A \times X$.

Recursion/Corecursion

Recursion principle

To define a function $u : \mathrm{List}(A) \to X$, it is enough to specify $u([])$ and to specify $u(\mathrm{cons}(a, \mathrm{as}))$ in terms of a and $u(\mathrm{as})$.

Corecursion principle

To define a function $u : X \to \mathrm{coList}(A)$, it is enough to specify a partial function $f : X \nrightarrow A \times X$.

- $\mathrm{pop}(u(x)) = \mathrm{no}$ whenever $f(x) = \mathrm{no}$
- $\mathrm{pop}(u(x)) = (a, u(x'))$ whenever $f(x) = (a, x')$

Recursion/Corecursion

Recursion principle

To define a function $u : \mathrm{List}(A) \to X$, it is enough to specify $u([])$ and to specify $u(\mathrm{cons}(a, \mathrm{as}))$ in terms of a and $u(\mathrm{as})$.

Corecursion principle

To define a function $u : X \to \mathrm{coList}(A)$, it is enough to specify a partial function $f : X \nrightarrow A \times X$.

- $\mathrm{pop}(u(x)) = \mathrm{no}$ whenever $f(x) = \mathrm{no}$
- $\mathrm{pop}(u(x)) = (a, u(x'))$ whenever $f(x) = (a, x')$

Think of $u(x) = [a_0, a_1, \ldots]$ as tracing the execution

$$x \overset{f}{\mapsto} (a_0, x') \overset{f}{\mapsto} (a_0, (a_1, x'')) \overset{f}{\mapsto} \cdots$$

Induction/Coinduction

Induction principle

Let $\mathrm{lhs}, \mathrm{rhs} : \mathrm{List}(A) \to X$. If

- $\mathrm{lhs}([]) = \mathrm{rhs}([])$
- $\mathrm{lhs}(\mathrm{cons}(a, \mathrm{as})) = \mathrm{rhs}(\mathrm{cons}(a, \mathrm{as}))$ whenever $\mathrm{lhs}(\mathrm{as}) = \mathrm{rhs}(\mathrm{as})$

then $\mathrm{lhs} = \mathrm{rhs}$.

Induction/Coinduction

Induction principle

Let $\mathrm{lhs}, \mathrm{rhs} : \mathrm{List}(A) \to X$. If

- $\mathrm{lhs}([]) = \mathrm{rhs}([])$
- $\mathrm{lhs}(\mathrm{cons}(a, \mathrm{as})) = \mathrm{rhs}(\mathrm{cons}(a, \mathrm{as}))$ whenever $\mathrm{lhs}(\mathrm{as}) = \mathrm{rhs}(\mathrm{as})$

then $\mathrm{lhs} = \mathrm{rhs}$.

Example theorems:

- Concatenation of lists is associative
- Length of concatenation is sum of lengths $(\text{induct from List})$
- List is a functor
- List has the "list comprehension" applicative from Haskell

Coinduction principle

Let $\mathrm{lhs}, \mathrm{rhs} : X \to \mathrm{coList}(A)$.
Let $[-] : \mathrm{coList}(a) \to \mathrm{coList}(A)_{/\mathrm{lhs}=\mathrm{rhs}}$ be the quotient map.
If

$$\forall x \in X,\ [\mathrm{pop}(\mathrm{lhs}(x))] = [\mathrm{pop}(\mathrm{rhs}(x))]$$

(with $[\mathrm{no}] := \mathrm{no}$), then $\mathrm{lhs} = \mathrm{rhs}$.

Induction/Coinduction

Coinduction principle

Let $\mathrm{lhs}, \mathrm{rhs} : X \to \mathrm{coList}(A)$.
Let $[-] : \mathrm{coList}(a) \to \mathrm{coList}(A)_{/\mathrm{lhs}=\mathrm{rhs}}$ be the quotient map.
If

$$\forall x \in X,\ [\mathrm{pop}(\mathrm{lhs}(x))] = [\mathrm{pop}(\mathrm{rhs}(x))]$$

(with $[\mathrm{no}] := \mathrm{no}$), then $\mathrm{lhs} = \mathrm{rhs}$.

Example theorems:

- Concatenation of colists is associative
- Length of concatenation is sum of lengths (coinduct into $\mathrm{co}\mathbb{N}$)
- coList is a functor
- coList has the "ZipList" applicative from Haskell

Other structures

Every (non-dependent) inductive `data` you can make in Haskell/Lean/Agda/etc. has a coinductive version.

Other structures

Every (non-dependent) inductive `data` you can make in Haskell/Lean/Agda/etc. has a coinductive version.

- $\mathbb{N} \rightsquigarrow \mathrm{co}\mathbb{N}$

Other structures

Every (non-dependent) inductive `data` you can make in Haskell/Lean/Agda/etc. has a coinductive version.

- $\mathbb{N} \rightsquigarrow \mathrm{co}\mathbb{N}$
- $\mathrm{List} \rightsquigarrow \mathrm{coList}$

Other structures

Every (non-dependent) inductive `data` you can make in Haskell/Lean/Agda/etc. has a coinductive version.

- $\mathbb{N} \rightsquigarrow \mathrm{co}\mathbb{N}$
- $\mathrm{List} \rightsquigarrow \mathrm{coList}$
- Finite-size binary trees $\rightsquigarrow$ potentially infinite binary trees

Other structures

Every (non-dependent) inductive `data` you can make in Haskell/Lean/Agda/etc. has a coinductive version.

- $\mathbb{N} \rightsquigarrow \mathrm{co}\mathbb{N}$
- $\mathrm{List} \rightsquigarrow \mathrm{coList}$
- Finite-size binary trees $\rightsquigarrow$ potentially infinite binary trees
- Etc.

Outline

All the categories

All of the inductive `data` types are **initial algebras**.

Initial algebras

Let $T : \mathcal{C} \to \mathcal{C}$ be an endofunctor.

All the categories

All of the inductive `data` types are **initial algebras**.

Initial algebras

Let $T : \mathcal{C} \to \mathcal{C}$ be an endofunctor.

An **algebra** for T is an arrow $a \xrightarrow{\alpha} T(a)$ in $\mathcal{C}$.

All the categories

All of the inductive `data` types are **initial algebras**.

Initial algebras

Let $T : \mathcal{C} \to \mathcal{C}$ be an endofunctor.

An **algebra** for T is an arrow $a \xrightarrow{\alpha} T(a)$ in $\mathcal{C}$.

A **morphism** from $a \xrightarrow{\alpha} T(a)$ to $b \xrightarrow{\beta} T(b)$ is an arrow $f : a \to b$ in $\mathcal{C}$ such that this commutes:

$$\begin{tikzcd} a \arrow[r, "\alpha"] \arrow[d, "f"'] & T(a) \arrow[d, "T(f)"] \\ b \arrow[r, "\beta"'] & T(b) \end{tikzcd}$$

All the categories

All of the inductive `data` types are **initial algebras**.

Initial algebras

Let $T : \mathcal{C} \to \mathcal{C}$ be an endofunctor.

An **algebra** for T is an arrow $a \xrightarrow{\alpha} T(a)$ in $\mathcal{C}$.

A **morphism** from $a \xrightarrow{\alpha} T(a)$ to $b \xrightarrow{\beta} T(b)$ is an arrow $f : a \to b$ in $\mathcal{C}$ such that this commutes:

$$\begin{tikzcd} a \arrow[r, "\alpha"] \arrow[d, "f"'] & T(a) \arrow[d, "T(f)"] \\ b \arrow[r, "\beta"'] & T(b) \end{tikzcd}$$

With identity and composite arrows as in $\mathcal{C}$, these form a category.

All the categories

All of the inductive `data` types are **initial algebras**.

Initial algebras

Let $T : \mathcal{C} \to \mathcal{C}$ be an endofunctor.

An **algebra** for T is an arrow $a \xrightarrow{\alpha} T(a)$ in $\mathcal{C}$.

A **morphism** from $a \xrightarrow{\alpha} T(a)$ to $b \xrightarrow{\beta} T(b)$ is an arrow $f : a \to b$ in $\mathcal{C}$ such that this commutes:

$$
\begin{tikzcd}
a \arrow[r, "\alpha"] \arrow[d, "f"'] & T(a) \arrow[d, "T(f)"] \\
b \arrow[r, "\beta"'] & T(b)
\end{tikzcd}
$$

With identity and composite arrows as in $\mathcal{C}$, these form a category.

An **initial algebra** for T is an initial object in this category.

Recursion/Induction

All of the inductive `data` types are **initial algebras**.

Recursion/Induction

All of the inductive `data` types are **initial algebras**.

Recursion

An **initial algebra** for T is an initial object $i \xrightarrow{\text{cons}} T(i)$ in the category of T-algebras.

Recursion/Induction

All of the inductive `data` types are **initial algebras**.

Recursion

An **initial algebra** for T is an initial object $i \xrightarrow{\text{cons}} T(i)$ in the category of T-algebras.

The **recursion principle** is its universal property.

Induction

Let $i \xrightarrow{\text{cons}} T(i)$ be an initial algebra.
Let $f, g : i \to a$ in $\mathcal{C}$.
Suppose the equaliser $\iota : e \rightarrowtail i$ of f and g exists in $\mathcal{C}$, and that $f \circ \text{cons} \circ T(\iota) = g \circ \text{cons} \circ T(\iota)$:

$$
\begin{tikzcd}
T(e) \arrow[r, dashed] \arrow[d, "T(\iota)"'] & e \arrow[d, tail, "\iota"] \\
T(i) \arrow[r, "\text{cons}"] & i \arrow[d, shift right, "f"'] \arrow[d, shift left, "g"] \\
 & a
\end{tikzcd}
$$

Then, $f = g$.

Corecursion/Coinduction

Turn all the arrows around.

Corecursion/Coinduction

Turn all the arrows around.

Coinduction

Let $T(j) \xrightarrow{\text{des}} j$ be an initial algebra.
Let $f, g : a \to j$ in $\mathcal{C}$.
Suppose the coequaliser $\pi : j \twoheadrightarrow e$ of f and g exists in $\mathcal{C}$, and that this commutes:

$$\begin{tikzcd}
a \arrow[d, shift right, "f"'] \arrow[d, shift left, "g"] & \\
j \arrow[r, "\text{des}"] \arrow[d, two heads, "\pi"'] & T(j) \arrow[d, "T(\pi)"] \\
e \arrow[r, dashed] & T(e)
\end{tikzcd}$$

Then, $f = g$.

Something cool ig

$T : \mathbf{Set} \to \mathbf{Set}$ given by $X \mapsto \{\mathrm{no}\} \sqcup X$ has:

- Initial algebra $\{\mathrm{no}\} \sqcup \mathbb{N} \xrightarrow{(0,\mathrm{succ})} \mathbb{N}$.
- Terminal coalgebra $\mathbb{N} \xrightarrow{\mathrm{pred}} \{\mathrm{no}\} \sqcup \mathbb{N}$.

Something cool ig

$T : \mathbf{Set} \to \mathbf{Set}$ given by $X \mapsto \{\mathrm{no}\} \sqcup X$ has:

- Initial algebra $\{\mathrm{no}\} \sqcup \mathbb{N} \xrightarrow{(0,\mathrm{succ})} \mathbb{N}$.
- Terminal coalgebra $\mathbb{N} \xrightarrow{\mathrm{pred}} \{\mathrm{no}\} \sqcup \mathbb{N}$.

Instead regard $T : \mathbf{Top} \to \mathbf{Top}$.

Something cool ig

$T : \mathbf{Set} \to \mathbf{Set}$ given by $X \mapsto \{\text{no}\} \sqcup X$ has:

- Initial algebra $\{\text{no}\} \sqcup \mathbb{N} \xrightarrow{(0,\text{succ})} \mathbb{N}$.
- Terminal coalgebra $\mathbb{N} \xrightarrow{\text{pred}} \{\text{no}\} \sqcup \mathbb{N}$.

Instead regard $T : \mathbf{Top} \to \mathbf{Top}$.

- The initial algebra exists, and is $\mathbb{N}$ with discrete topology.

Something cool ig

$T : \mathbf{Set} \to \mathbf{Set}$ given by $X \mapsto \{\text{no}\} \sqcup X$ has:

- Initial algebra $\{\text{no}\} \sqcup \mathbb{N} \xrightarrow{(0,\text{succ})} \mathbb{N}$.
- Terminal coalgebra $\mathbb{N} \xrightarrow{\text{pred}} \{\text{no}\} \sqcup \mathbb{N}$.

Instead regard $T : \mathbf{Top} \to \mathbf{Top}$.

- The initial algebra exists, and is $\mathbb{N}$ with discrete topology.
- The terminal coalgebra exists, and is $\text{co}\mathbb{N}$ with

$$(0, 1, 2, \dots) \xrightarrow{\text{converges}} \infty$$