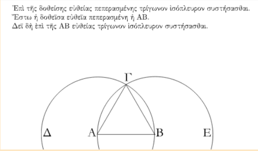
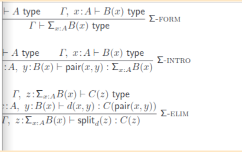
Matemática sintética e teoria dos tipos

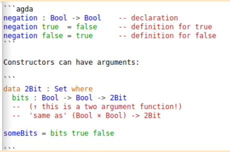




$$\frac{\vdash A \text{ type} \qquad \Gamma,\ x{:}A \vdash B(x) \text{ type}}{\Gamma \vdash \Sigma_{x:A} B(x) \text{ type}}\ \Sigma\text{-FORM}$$

$$\frac{\vdash A \text{ type} \qquad \Gamma,\ x{:}A \vdash B(x) \text{ type}}{:A,\ y{:}B(x) \vdash \mathsf{pair}(x,y) : \Sigma_{x:A} B(x)}\ \Sigma\text{-INTRO}$$

$$\frac{\begin{array}{c}\Gamma,\ z{:}\Sigma_{x:A} B(x) \vdash C(z) \text{ type} \\ :A,\ y{:}B(x) \vdash d(x,y) : C(\mathsf{pair}(x,y))\end{array}}{\Gamma,\ z{:}\Sigma_{x:A} B(x) \vdash \mathsf{split}_d(z) : C(z)}\ \Sigma\text{-ELIM}$$



```
```agda
negation : Bool -> Bool    -- declaration
negation true  = false     -- definition for true
negation false = true      -- definition for false
```

Constructors can have arguments:

```
data 2Bit : Set where
  bits : Bool -> Bool -> 2Bit
  --  (↑ this is a two argument function!)
  --  'same as' (Bool × Bool) -> 2Bit

someBits = bits true false

```
```

Peter Arndt

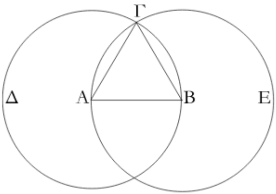
UFABC, 20 Março 2024

Matemática sintética versus matemática analítica

Matemática sintética versus matemática analítica

Matemática sintética:

Ἐπὶ τῆς δοθείσης εὐθείας πεπερασμένης τρίγωνον ἰσόπλευρον συστήσασθαι.
Ἔστω ἡ δοθεῖσα εὐθεῖα πεπερασμένη ἡ ΑΒ.
Δεῖ δὴ ἐπὶ τῆς ΑΒ εὐθείας τρίγωνον ἰσόπλευρον συστήσασθαι.



Κέντρῳ μὲν τῷ Α διαστήματι δὲ τῷ ΑΒ κύκλος γεγράφθω ὁ ΒΓΔ, καὶ πάλιν κέντρῳ μὲν τῷ Β διαστήματι δὲ τῷ ΒΑ κύκλος γεγράφθω ὁ ΑΓΕ, καὶ ἀπὸ τοῦ Γ σημείου, καθ᾽ ὃ τέμνουσιν ἀλλήλους οἱ κύκλοι, ἐπί τὰ Α, Β σημεῖα ἐπεζεύχθωσαν εὐθεῖαι αἱ ΓΑ, ΓΒ.

Καὶ ἐπεὶ τὸ Α σημεῖον κέντρον ἐστὶ τοῦ ΓΔΒ κύκλου, ἴση ἐστὶν ἡ ΑΓ τῇ ΑΒ· πάλιν, ἐπεὶ τὸ Β σημεῖον κέντρον ἐστὶ τοῦ ΓΑΕ κύκλου, ἴση ἐστὶν ἡ ΒΓ τῇ ΒΑ. ἐδείχθη δὲ καὶ ἡ ΓΑ τῇ ΑΒ ἴση· ἑκατέρα ἄρα τῶν ΓΑ, ΓΒ τῇ ΑΒ ἐστιν ἴση. τὰ δὲ τῷ αὐτῷ ἴσα καὶ ἀλλήλοις ἐστὶν ἴσα· καὶ ἡ ΓΑ ἄρα τῇ ΓΒ ἐστιν ἴση· αἱ τρεῖς ἄρα αἱ ΓΑ, ΑΒ, ΒΓ ἴσαι ἀλλήλαις εἰσίν.

Ἰσόπλευρον ἄρα ἐστὶ τὸ ΑΒΓ τρίγωνον. καὶ συνέσταται ἐπὶ τῆς δοθείσης εὐθείας πεπερασμένης τῆς ΑΒ. ὅπερ ἔδει ποιῆσαι.

Matemática sintética versus matemática analítica

Matemática sintética:

Geometria plana

P pontos

R retas

vocabulário / estrutura

$$\text{„}\parallel\text{"} \subseteq R \times R$$

$$P \times P \setminus \{(p,p) \mid p \in P\} \xrightarrow{\mathcal{R}} R$$

$$R \times R \setminus \{(r,r') \mid r \parallel r'\} \xrightarrow{\mathcal{I}} P$$

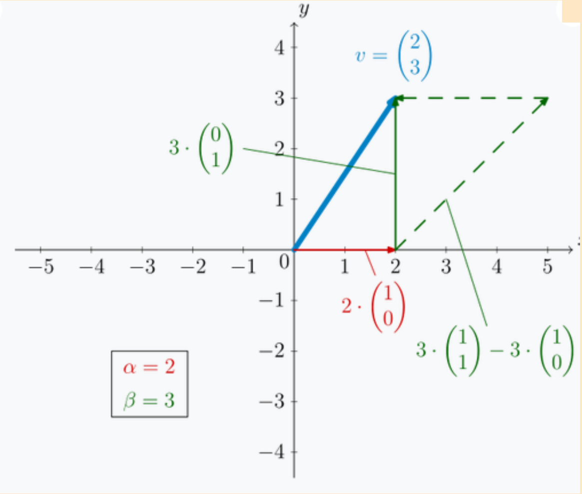
axiomas

$r \parallel r'$ e $p = \mathcal{I}(r,r'')$, $p' = \mathcal{I}(r',r'')$

então $p \neq p'$ ...

Matemática sintética versus matemática analítica

Matemática analítica:



Matemática sintética versus matemática analítica

Matemática analítica:

$$P := \left\{ \begin{pmatrix} a \\ b \end{pmatrix} \;\middle|\; a, b \in \mathbb{R} \right\}$$

$$R := \left\{ \left\{ \begin{pmatrix} a \\ b \end{pmatrix} + \lambda \begin{pmatrix} c \\ d \end{pmatrix} \;\middle|\; \lambda \in \mathbb{R} \right\} \;\middle|\; a, b, c, d \in \mathbb{R} \right\}$$

$$r \parallel r' :\Longleftrightarrow \begin{array}{l} r = \left\{ \begin{pmatrix} a \\ b \end{pmatrix} + \lambda \begin{pmatrix} c \\ d \end{pmatrix} \;\middle|\; \lambda \in \mathbb{R} \right\} \\ r' = \left\{ \begin{pmatrix} a' \\ b' \end{pmatrix} + \lambda \begin{pmatrix} c' \\ d' \end{pmatrix} \;\middle|\; \lambda \in \mathbb{R} \right\} \end{array} \Rightarrow \begin{pmatrix} c \\ d \end{pmatrix} = \mu \cdot \begin{pmatrix} c' \\ d' \end{pmatrix} \text{ para um } \mu \in \mathbb{R}$$

$$\begin{aligned} P \times P \setminus \left\{ \left(\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} a \\ b \end{pmatrix} \right) \;\middle|\; a, b \in \mathbb{R} \right\} &\xrightarrow{\quad \mathcal{R} \quad} R \\ \left(\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} a' \\ b' \end{pmatrix} \right) &\longmapsto \left\{ \begin{pmatrix} a \\ b \end{pmatrix} + \lambda \begin{pmatrix} a - a' \\ b - b' \end{pmatrix} \;\middle|\; \lambda \in \mathbb{R} \right\} \end{aligned}$$

Matemática sintética versus matemática analítica

Matemática sintética:

Toma os objetos do discurso como primitivos

Axiomatiza o comportamento

Matemática analítica:

Define os objetos em termos de outros objetos mais básicos

Prova as propriedades do comportamento

Analogia com programação

Matemática sintética

↕

linguagens domain specific (DSLs)

- simples,
- vocabulário restrito,
- não Turing completas
- accessível p/ não expertos

por ex: html, SQL, TeX

Matemática analítica

↕

linguagens gerais

- complexas
- grande vocabulário
- Turing completas
- p/ „especialistas" de programação

C, Java, Python

Analogia com programação

Matemática sintética $\xrightarrow{\text{modelagem}}$ **Matemática analítica**

linguagens domain specific (DSLs) $\xrightarrow{\text{Compilação /interpretação}}$ **linguagens gerais**

- simples,
- vocabulário restrito,
- não Turing completas
- accessível p/ não expertos

por ex: html, SQL, TeX

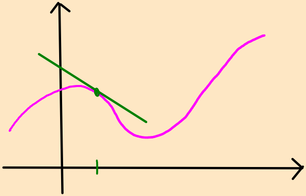
- complexas
- grande vocabulário
- Turing completas
- p/ „especialistas" de programação

C, Java, Python

Análise sintética

Cálculo: Derivada de uma função

$$f : \mathbb{R} \longrightarrow \mathbb{R}$$



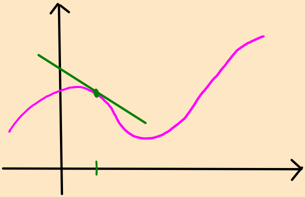
Approximação local linear

$$f(x+\varepsilon) \approx f(x) + \underbrace{b}_{f'(x)} \cdot \varepsilon \quad \text{p/ } \varepsilon \text{ pequeno}$$

Análise sintética

Cálculo: Derivada de uma função

$$f : \mathbb{R} \longrightarrow \mathbb{R}$$



Approximação local linear

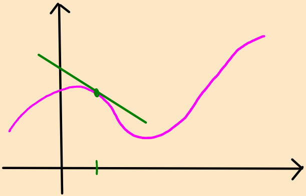
$$f(x+\varepsilon) \approx f(x) + \underbrace{b}_{f'(x)} \cdot \varepsilon \quad \text{p/ } \varepsilon \text{ pequeno}$$

i.e. $f(x+\varepsilon) - \left(f(x) + b \cdot \varepsilon\right) \in o(\varepsilon)$

Análise sintética

Cálculo: Derivada de uma função

$$f: \mathbb{R} \longrightarrow \mathbb{R}$$



Approximação local linear

$$f(x+\varepsilon) \approx f(x) + \underbrace{b}_{f'(x)} \cdot \varepsilon$$ p/ ε pequeno

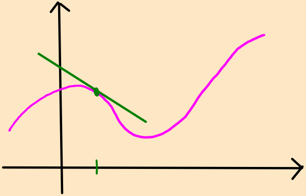
i.e. $f(x+\varepsilon) - \left(f(x) + b \cdot \varepsilon\right) \in o(\varepsilon)$

ou $b = \lim\limits_{\varepsilon \to 0} \dfrac{f(x+\varepsilon) - f(x)}{\varepsilon}$

Análise sintética

Cálculo: Derivada de uma função

$$f : \mathbb{R} \longrightarrow \mathbb{R}$$



Approximação local linear

$$f(x+\varepsilon) \approx f(x) + \underbrace{b}_{f'(x)} \cdot \varepsilon$$

p/ ε pequeno

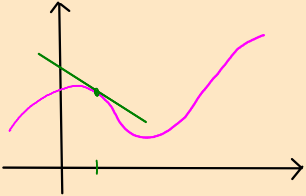
Se $f(x+z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + \dots$

e $\varepsilon^2 = 0$, então $f(x+\varepsilon) = f(x) + b \cdot \varepsilon$

Análise sintética

Cálculo: Derivada de uma função

$$f : \mathbb{R} \longrightarrow \mathbb{R}$$



Approximação local linear

$$f(x+\varepsilon) \approx f(x) + \underbrace{b}_{f'(x)} \cdot \varepsilon$$

p/ ε pequeno

Se $f(x+z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + \dots$

e $\varepsilon^2 = 0$, então $f(x+\varepsilon) = f(x) + b \cdot \varepsilon$

Axioma de Kock-Lawvere:

Para todo $f: \mathbb{R} \longrightarrow \mathbb{R}$ e $x \in \mathbb{R}$ existe um unico $b \in \mathbb{R}$ tal que, para todo ε com $\varepsilon^2 = 0$,

$$f(x+\varepsilon) = f(x) + b \cdot \varepsilon$$

Axioma de Kock-Lawvere:

Para todo $f: \mathbb{R} \to \mathbb{R}$ e $x \in \mathbb{R}$ existe um unico $b \in \mathbb{R}$ tal que, para todo ε com $\varepsilon^2 = 0$,

$$f(x+\varepsilon) = f(x) + b \cdot \varepsilon$$

$$\Delta := \{\varepsilon \in \mathbb{R} \mid \varepsilon^2 = 0\}$$

Toda função $\Delta \xrightarrow{g} \mathbb{R}$ é da forma $\varepsilon \mapsto g(0) + b \cdot \varepsilon$ para um unico b.

Axioma de Kock-Lawvere:

Para todo $f : \mathbb{R} \to \mathbb{R}$ e $x \in \mathbb{R}$ existe um unico $b \in \mathbb{R}$ tal que, para todo ε com $\varepsilon^2 = 0$,

$$f(x+\varepsilon) = f(x) + b \cdot \varepsilon$$

$$\Delta := \{ \varepsilon \in \mathbb{R} \mid \varepsilon^2 = 0 \}$$

Toda função $\Delta \xrightarrow{g} \mathbb{R}$ é da forma $\varepsilon \mapsto g(0) + b \cdot \varepsilon$ para um unico b.

$$[\, g(\varepsilon) := f(x+\varepsilon) \,]$$

Axioma de Kock-Lawvere:

Para todo $f: \mathbb{R} \longrightarrow \mathbb{R}$ e $x \in \mathbb{R}$ existe um unico $b \in \mathbb{R}$ tal que, para todo ε com $\varepsilon^2 = 0$,

$$f(x+\varepsilon) = f(x) + b \cdot \varepsilon$$

$$\Delta := \{ \varepsilon \in \mathbb{R} \mid \varepsilon^2 = 0 \}$$

Toda função $\Delta \xrightarrow{g} \mathbb{R}$ é da forma $\varepsilon \longmapsto g(0) + b \cdot \varepsilon$ para um unico b.

„Δ é tão pequeno que não tem espaço suficiente para a reta se dobrar."

Análise sintética

Derivada de $f(x) = x^3$:

$$f(x+\varepsilon) = (x+\varepsilon)^3 = \underbrace{x^3}_{f(x)} + \underbrace{3x^2}_{f'(x)} \cdot \varepsilon + \underbrace{3x\varepsilon^2 + \varepsilon^3}_{=0}$$

Análise sintética

Derivada de $f(x) = x^3$:

$$f(x+\varepsilon) = (x+\varepsilon)^3 = \underbrace{x^3}_{f(x)} + \underbrace{3x^2}_{f'(x)} \cdot \varepsilon + \underbrace{3x\varepsilon^2 + \varepsilon^3}_{=0}$$

Regra do produto:

$$(f \cdot g)(x+\varepsilon) = f(x+\varepsilon) \cdot g(x+\varepsilon)$$
$$= \left(f(x) + f'(x) \cdot \varepsilon\right)\left(g(x) + g'(x) \cdot \varepsilon\right)$$
$$= \underbrace{f(x) \cdot g(x)}_{(f \cdot g)(x)} + \underbrace{\left(f'(x) \cdot g(x) + f(x) \cdot g'(x)\right)}_{(f \cdot g)'(x)} \cdot \varepsilon + \underbrace{f'(x) g'(x) \cdot \varepsilon^2}_{=0}$$

Analogia com programação

High level language

por ex. Python, Java,
...

tem pacotes,
garbage collection,
sintaxe legivel

Low level language

Assembler,
ByteCode,
C

tem que
reservar memoria,
implementar estr.
de dados etc.

Analogia com programação

High level language $\xrightarrow{\text{compila}}$ **Low level language**

por ex. Python, Java,
...

Assembler,
ByteCode,
C

tem pacotes,
garbage collection,
sintaxe legível

tem que
reservar memória,
implementar estr.
de dados etc.

Análise sintética

Toda função $\Delta \xrightarrow{g} \mathbb{R}$ é da forma $\varepsilon \longmapsto g(0) + b \cdot \varepsilon$ para um unico $\underset{f'(x)}{\overset{\|}{b}}$.

Se $g(\varepsilon) = f(x+\varepsilon) \rightsquigarrow f'(x)$

$f' : \mathbb{R} \to \mathbb{R}$ de novo diferenciavel ...

$\rightsquigarrow$ Toda função $\mathbb{R} \to \mathbb{R}$ é lisa.

Análise sintética

Clássicamente: $\Delta = \{\varepsilon \in \mathbb{R} \mid \varepsilon^2 = 0\} = \{0\}$

Mas o axioma de Kock-Lawvere contradiz isto:

Se $\Delta = \{0\}$, então $f(x+\varepsilon) = f(x) + b \cdot \varepsilon$

para <u>todo</u> b, mas o b é único.

Análise sintética

Clássicamente: $\Delta = \{\varepsilon \in \mathbb{R} \mid \varepsilon^2 = 0\} = \{0\}$

Mas o axioma de Kock-Lawvere contradiz isto:

Se $\Delta = \{0\}$, então $f(x+\varepsilon) = f(x) + b \cdot \varepsilon$ para <u>todo</u> b, mas o b é único.

Kock-Lawvere $\Longrightarrow$ $\Delta \neq \{0\}$

Análise sintética

Kock-Lawvere $\Longrightarrow$ $\Delta \neq \{0\}$

Considere $$\begin{aligned} f : \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto \begin{cases} 0 & x = 0 \\ 1 & x \neq 0 \end{cases} \end{aligned}$$

Então $$f(0+\varepsilon) = f'(0)\cdot\varepsilon = \begin{cases} 0 & \text{se } \varepsilon = 0 \\ 1 & \text{se } \varepsilon \neq 0 \end{cases}$$

Para $\varepsilon \neq 0$: $$\varepsilon = 1\cdot\varepsilon = (f'(0)\cdot\varepsilon)\cdot\varepsilon = \underbrace{f'(0)\cdot\varepsilon^2}_{= 0}$$

$$\Longrightarrow \Delta = \{0\} \quad \lightning$$

Análise sintética

Kock-Lawvere $\Longrightarrow$ $\Delta \neq \{0\}$

Co... $f : \mathbb{R} \to \mathbb{R}$

$$\begin{cases} 0 & x = 0 \\ 1 & x \neq 0 \end{cases}$$

Ent... $(0) \cdot \varepsilon = \begin{cases} 0 & \text{se } \varepsilon = 0 \\ 1 & \text{se } \varepsilon \neq 0 \end{cases}$

Pa... $1 \cdot \varepsilon = (f'(0) \cdot \varepsilon) \cdot \varepsilon = f'(0) \cdot \varepsilon^2 = 0$

$\Longrightarrow$ $\Delta = \{0\}$ $\lightning$

Análise sintética

O axioma de Kock-Lawvere é incompatível com a lógica clássica

— mas não com a lógica construtiva!

Análise sintética

O axioma de Kock-Lawvere é incompatível com a lógica clássica

— mas não com a lógica construtiva!

Lá não vale $A \vee \neg A$

Análise sintética

O axioma de Kock-Lawvere é incompatível com a lógica clássica

— mas não com a lógica construtiva!

La não vale $A \vee \neg A$

$\rightsquigarrow \quad \mathbb{R} \neq \{x \in \mathbb{R} \mid x = 0 \vee x \neq 0\}$

$\rightsquigarrow \quad f : x \longmapsto \begin{cases} 0 & \text{se } x = 0 \\ 1 & \text{se } x \neq 0 \end{cases}$

não é uma função $\mathbb{R} \to \mathbb{R}$

Analogia com programação

High level language $\xrightarrow[\text{exigente}]{\text{compilador}}$ **Low level language**

por ex. ~~Python~~, ~~Java~~,
Haskell, ML

Assembler,
ByteCode,
C

tem pacotes,
garbage collection,
sintaxe legível
sistema de tipos
rígido.

tem que
reservar memoria,
implementar estr.
de dados etc.

lógica construtiva

$\mathbb{R}$ é um corpo: Vale $x \neq 0 \Rightarrow \exists y : x \cdot y = 1$

$\mathbb{R}$ não é um corpo': não vale $x = 0 \vee \exists y : x \cdot y = 1$.

Provamos $\Delta \neq \{0\}$, mas para nenhum $\varepsilon \in \Delta$ podemos provar que $\varepsilon \neq 0$:

Análise sintética

lógica construtiva

(*)

$\mathbb{R}$ é um corpo: Vale $\boxed{x \neq 0 \Rightarrow \exists y: x \cdot y = 1}$

$\mathbb{R}$ não é um corpo!: não vale $x = 0 \vee \exists y: x \cdot y = 1$.

Provamos $\Delta \neq \{0\}$, mas para nenhum $\varepsilon \in \Delta$ podemos provar que $\varepsilon \neq 0$:

$$\varepsilon \neq 0 \Rightarrow \varepsilon = \varepsilon \cdot 1 = \varepsilon\left(\varepsilon \cdot \frac{1}{\varepsilon}\right) = \varepsilon^2 \cdot \frac{1}{\varepsilon} = 0 \cdot \frac{1}{\varepsilon} = 0$$

(*) ↯

Análise sintética

Sumário:

- **todas as funções são lisas**
- **toda função $\Delta \to \mathbb{R}$ é linear**
- **$\mathbb{R}$ é um corpo**

Análise sintética

Sumário:
- **todas as funções são lisas**
- **toda função $\Delta \to \mathbb{R}$ é linear**
- **$\mathbb{R}$ é um corpo**

Geometria diferencial sintética

M variedade lisa $\rightsquigarrow$ $TM := M^\Delta = \{\Delta \to M\}$

$$T_xM := \{ t : \Delta \to M \mid t(0) = x \}$$

$\rightsquigarrow$ formas diferenciais, fibrados, conexões, álgebras de Lie, jatos, Gauss-Bonnet, Cohomologia de de Rham ...

Outras abordagens sintéticas

Topologia sintética:

Todas as funções são continuas

Computabilidade sintética:

Todas as funções são computáveis

Teoria sintética dos domínios:

Todas as funções preservam ordem (e mais)

Probabilidade sintética:

Todas as funções são mapas estocásticos

Geometria algébrica sintética:

Todas as funções são polinomiais

Analogia com programação

Matemática sintética $\overset{?}{\rightsquigarrow}$ **Matemática analítica**

$\updownarrow$ $\updownarrow$

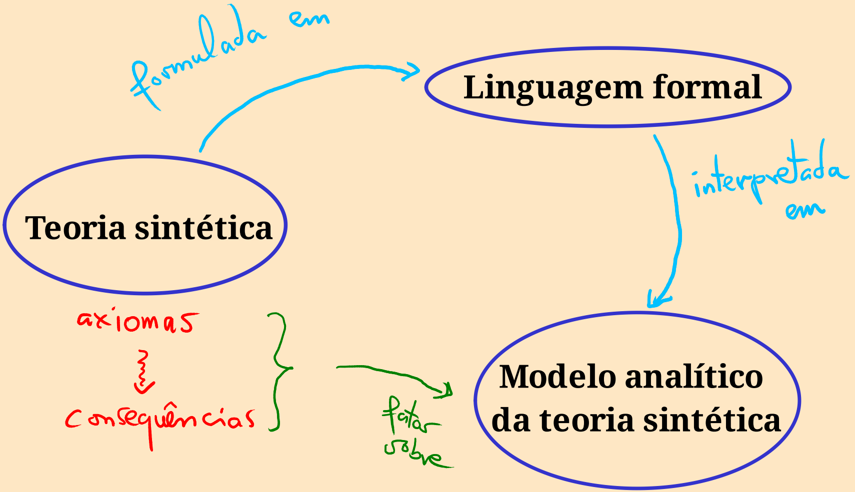
linguagens domain specific (DSLs) $\xrightarrow[\text{/interpretação}]{\text{compilação}}$ **linguagens gerais**

- simples,
- vocabulário restrito,
- não Turing completas
- accessível p/ não expertos

- complexas
- grande vocabulário
- Turing completas
- p/ „especialistas" de programação

por ex: html, SQL, .. C, Java, Python, ...

Ligação com a matemática analítica



Ligação com a matemática analítica

Exemplo: Geometria diferêncial sintética

$\forall, \exists, \wedge, \vee \rightarrow, \neg$

formulada em

Linguagem 1a ordem

interpretada em

Geom. dif. sintética

$$\forall f \in \mathbb{R}^{\mathbb{R}}, x \in \mathbb{R}, \exists! b \in \mathbb{R} \forall \epsilon \in \Delta : f(x+\epsilon) = f(x) + b \cdot \epsilon$$

$$\Delta \neq \{0\}$$

Topos

$$Set^{Var^{op}}$$

$$\mathcal{T}$$

Ligação com a matemática analítica

Exemplo: Geometria diferêncial sintética

$\forall, \exists, \wedge, \vee \rightarrow, \neg$

formulada em

Linguagem 1a ordem

interpretada em

Geom. dif. sintética

$\forall f \in \mathbb{R}^{\mathbb{R}}, x \in \mathbb{R}, \exists! b \in \mathbb{R} \forall \epsilon \in \Delta : f(x+\epsilon) = f(x) + b \cdot \epsilon$

$\Delta \neq \{0\}$

Topos

$Set^{Var^{op}}$

$\mathcal{T}$

$\mathbb{R} : Var^{op} \rightarrow Set,$

$M \mapsto C^{\infty}(M, \mathbb{R})$

Uma linguagem mais expressiva

$\forall, \exists, \wedge, \vee \to, \neg$

Linguagem 1a ordem

correspondência de Curry-Howard

$Id, \Pi, \Sigma, \times, \coprod, \to, \perp$

Teoria dos Tipos

interpretada em

∞-topos

Teoria dos Tipos

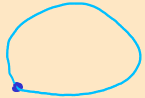
$$\frac{\Gamma \vdash A \ \mathsf{type} \qquad \Gamma,\ x{:}A \vdash B(x)\ \mathsf{type}}{\Gamma \vdash \Sigma_{x:A} B(x)\ \mathsf{type}}\ \Sigma\text{-}\textsc{form}$$

$$\frac{\Gamma \vdash A \ \mathsf{type} \qquad \Gamma,\ x{:}A \vdash B(x)\ \mathsf{type}}{\Gamma,\ x{:}A,\ y{:}B(x) \vdash \mathsf{pair}(x,y) : \Sigma_{x:A} B(x)}\ \Sigma\text{-}\textsc{intro}$$

$$\frac{\begin{array}{c}\Gamma,\ z{:}\Sigma_{x:A} B(x) \vdash C(z)\ \mathsf{type} \\ \Gamma,\ x{:}A,\ y{:}B(x) \vdash d(x,y) : C(\mathsf{pair}(x,y))\end{array}}{\Gamma,\ z{:}\Sigma_{x:A} B(x) \vdash \mathsf{split}_d(z) : C(z)}\ \Sigma\text{-}\textsc{elim}$$

$$\frac{\begin{array}{c}\Gamma,\ z{:}\Sigma_{x:A} B(x) \vdash C(z)\ \mathsf{type} \\ \Gamma,\ x{:}A,\ y{:}B(x) \vdash d(x,y) : C(\mathsf{pair}(x,y))\end{array}}{\Gamma,\ x{:}A,\ y{:}B(x) \vdash \mathsf{split}_d(\mathsf{pair}(x,y)) = d(x,y) : C(\mathsf{pair}(x,y))}\ \Sigma\text{-}\textsc{comp}$$

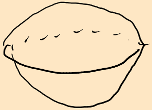
Interpretação homotopica



$x : A$

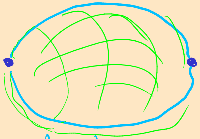
$c : \mathrm{Id}_A(x,x)$

$\rightsquigarrow$ teoria da homotopia sintética



$c : \mathrm{Id}_A(x,y)$

$x : A$



$y : A$

$d : \mathrm{Id}_A(x,y)$

$s : \mathrm{Id}_{\mathrm{Id}_A(x,y)}(c,d)$

$t : \mathrm{Id}_{\mathrm{Id}_A(x,y)}(c,d)$

Teoria da homotopia sintética

muito tecnico

Teoria da homotopia analítica

muito tecnico e removido da intuição

computações difíceis

Teoria da homotopia sintética	Teoria da homotopia analítica
muito tecnico – mas tem muitos usuários acustomados – tem ferramentas de software: Coq, Agda, HoTT-Lean	muito tecnico e removido da intuição computações difícis

Geometria algébrica sintética

https://github.com/felixwellen/synthetic-zariski/blob/main/README.md

Synthetic Algebraic Geometry in the Zariski-Topos

- Foundations (latest pdf, arxiv)
- Čech-Cohomology (early draft pdf)
- Differential Geometry/étale maps (early draft pdf)
- Proper Schemes (early draft pdf)
- Topology of Synthetic Schemes (early draft pdf)
- $\mathbb{A}^1$-homotopy theory (early draft pdf)
- Algebraic spaces and stacks (very early draft pdf)
- Automorphisms of and line bundles on projective space (very early draft pdf)
- More general topologies, in particular fppf (very early draft pdf)
- Calculations with (elliptic) curves and divisors (very early draft pdf)
- Synthetic stone duality (very early draft pdf)
- Finite schemes in SAG (very early draft pdf)
- Random Facts, i.e. a collection of everything that still needs to find a good place (very early draft pdf)
- Collection of exercises (pdf with exercise-ideas)

There is a related formalization project.

https://felix-cherubini.de/sag-meeting-4.html