Towards Modular Mathematics

Em-Cats
2022

Juan F. Meleiro
juan@juanmeleiro.mat.br
IME-USP
With support from CNPq
And advisorship by Hugo L. Mariano

Summary

1. Observations
2. Synthetic reasoning
3. Universal Logic
4. Formal framework
5. Uses

Observations

Some observations, values and desires:

- Assumption: Formalization is good.
- Observation: Everything is a model, and that is a good thing.
- Intuition: A model is a translation, and that simplifies stuff.
- Goal: Modularity; what? why?

Synthetic Reasoning
as a successful discipline

What is meant by **"synthetic" reasoning?** [...] It deals with space forms in therms of their structure, i.e. the basic geometric and conceptual constructions that can be performed on them. (**KockSynthetic1981**)

Generally, investigating geometric and quantitative relationships brings along with it understanding of the **logic appropriate for it**. (**KockSynthetic1981**)

Structure & Logic

Analytic vs. Synthetic: defining derivatives

- Do a bunch of set theory.
- Construct the real numbers $\mathbb{R}$.
- Define limits of a function $\mathbb{R} \to \mathbb{R}$

$$f(x) \xrightarrow{x \to x_0} c \iff$$

$$\forall \varepsilon > 0 \exists \delta > 0 \forall y \in \mathbb{R}(|x - x_0| < \delta \implies |f(x) - c| < \varepsilon)$$

- Define continuity of a function $\mathbb{R} \to \mathbb{R}$.

$$\forall x_0 \in \mathbb{R} \left(f(x) \xrightarrow{x \to x_0} f(x_0) \right)$$

- Define differentiability for continuous functions.

$$\forall x \in \mathbb{R} \exists c \in \mathbb{R} \left(\frac{f(x) - f(x+h)}{h} \xrightarrow{h \to 0} c \right)$$

- Do choice-woo to make that into a function.

Analytic vs. Synthetic: defining derivatives

- Do a bunch of set theory.
- Construct the real numbers $\mathbb{R}$.
- Define limits of a function $\mathbb{R} \to \mathbb{R}$

$$f(x) \xrightarrow{x \to x_0} c \iff$$

$$\forall \varepsilon > 0 \exists \delta > 0 \forall y \in \mathbb{R}(|x - x_0| < \delta \implies |f(x) - c| < \varepsilon)$$

- Define continuity of a function $\mathbb{R} \to \mathbb{R}$.

$$\forall x_0 \in \mathbb{R} \left(f(x) \xrightarrow{x \to x_0} f(x_0) \right)$$

- Define differentiability for continuous functions.

$$\forall x \in \mathbb{R} \exists c \in \mathbb{R} \left(\frac{f(x) - f(x+h)}{h} \xrightarrow{h \to 0} c \right)$$

- Do choice-woo to make that into a function.

- Declare signature: $(R, +, \cdot, 0, 1)$ is a ring.
- Define: $D := \{x : R \mid x^2 = 0\}$
- Postulate: Every $f : D \to R$ is of the form

$$f(d) = a + b \cdot d$$

(note that $a = f(0)$)

- Given $f : R \to R$ and $x : R$, apply the axiom to $\lambda d \,.\, f(x + d)$:

$$f(x + d) = f(x) + f'(x) \cdot d$$

Analytic vs. Synthetic: defining derivatives

- Do a bunch of set theory.
- Construct the real numbers $\mathbb{R}$.
- Define limits of a function $\mathbb{R} \to \mathbb{R}$

$$f(x) \xrightarrow{x \to x_0} c \iff$$
$$\forall \varepsilon > 0 \exists \delta > 0 \forall y \in \mathbb{R}(|x - x_0| < \delta \implies |f(x) - c| < \varepsilon)$$

- Define continuity of a function $\mathbb{R} \to \mathbb{R}$.

$$\forall x_0 \in \mathbb{R} \left(f(x) \xrightarrow{x \to x_0} f(x_0) \right)$$

- Define differentiability for continuous functions.

$$\forall x \in \mathbb{R} \exists c \in \mathbb{R} \left(\frac{f(x) - f(x+h)}{h} \xrightarrow{h \to 0} c \right)$$

- Do choice-woo to make that into a function.

- Declare signature: $(R, +, \cdot, 0, 1)$ is a ring.
- Define: $D := \{x : R \mid x^2 = 0\}$
- Postulate: Every $f : D \to R$ is of the form

$$f(d) = a + b \cdot d$$

  (note that $a = f(0)$)
- Given $f : R \to R$ and $x : R$, apply the axiom to $\lambda d \,.\, f(x + d)$:

$$f(x + d) = f(x) + f'(x) \cdot d$$

Done! Then... Given a space (type) M, a tangent vector at $p : M$ is a curve $t : D \to M$ with $t(0) = p$. The tangent bundle is M^D with projection π, and the tangent space T_pM is the fiber over p of π, *etc.*

From **KockMethods2016**,

- Every space M comes equipped with a reflexive symmetric relation $\sim_M$ (*e.g.*, $\sim : \prod_{M:\mathcal{U}} M \times M \to \mathcal{U}$).
- $x \sim_{k+1} y := \exists z \,.\, x \sim z \wedge z \sim_k y$.
- $\mathfrak{M}_k(x) := \{y : M \mid y \sim_k x\}$
- *etc*

> It is not the intention of SDG to avoid using the wonderful tool of coordinates. [...]
> The reason we did not start there, is to stress that the "arithmetization" in terms of [a ring] is a *tool*, not the *subject matter*, of geometry. [...] in particular, [geometry] has a life without the ring $\mathbb{R}$ of real numbers, who sometimes thinks of himself as being the owner and boss of the company. (Kock 2017)

Many ways to synthesize a subject (up to critical discussion)

From Bauer (2006),

- A type theory with types 1, sums, products, subsets, *etc*.
- Assume natural numbers $\mathbb{N}$.
- Define monotone binary sequences

$$\mathbb{N}^{+} := \{f : 2^{\mathbb{N}} \mid \forall n : \mathbb{N}\,(f(n) = 1 \to f(s(1)) = 1)\}$$

- Define several types of truth-values:
  - Standard: $\Omega := \mathcal{P}(1)$
  - Decidable: $2 := \{p : \Omega \mid p \vee \neg p\}$
  - Classical: $\Omega_{\neg\neg} := \{p : \Omega \mid \neg\neg p \to p\}$
- *etc*

In synthesis...

In sum:

- Synthetic reasoning involves structure and logic
- There is a rich field of possible "synthetizations" of an area

Issues:

- How can we combine and transform synthetic theories?

Universal Logic
as a conceptual framework

What logicians have to say:

> Logic is often informally described as the study of *sound reasoning*. [...] In an enormous development beginning in the late 19^{th} century, it has been found that a wide variety of different principles are needed for sound reasoning in different domains, and a "logic" has come to mean a set of principles for some form of sound reasoning. **But in a subject the essence of which is formalization, it is embarrassing that there is no widely acceptable formal definition of "a logic".** (Mossakowski et al. 2007, my emphasis)

The central questions

How to **identify**, **translate**, and **combine** logics?

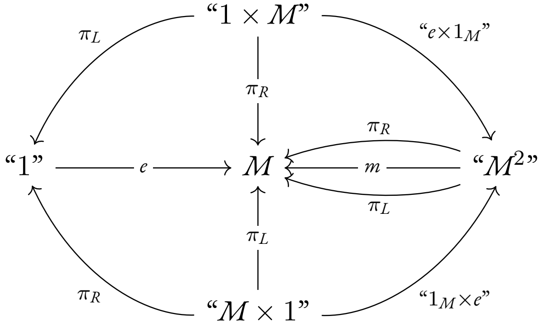
Categorical sketches

Categorical sketches are given by a diagram (category) $\mathcal{I}$, a set of cones $\mathcal{L}$ and a set of co-cones $\mathcal{C}$.

Categorical sketches are given by a diagram (category) $\mathcal{I}$, a set of cones $\mathcal{L}$ and a set of co-cones $\mathcal{C}$.

A model of a sketch $(\mathcal{I}, \mathcal{L}, \mathcal{C})$ in a category C is a functor $F : \mathcal{I} \to C$ taking cones in $\mathcal{L}$ to limits and co-cones in $\mathcal{C}$ to colimits.

A sketch for unital magmas[1] (nLab 2022) given by the category generated by



under some relations...

$$m \circ \text{“}e \times 1_M\text{”} = \pi_R$$
$$m \circ \text{“}1 \times e\text{”} = \pi_L$$
$$e \circ \pi_L = \pi_L \circ \text{“}e \times 1_M\text{”}$$
$$\pi_R \circ \text{“}e \times 1_M\text{”} = \pi_R$$
$$\pi_L = \text{“}1_M \times e\text{”} \circ \pi_L$$
$$e \circ \pi_R = \pi_R \circ \text{“}1_M \times e\text{”}$$

and cones making the π’s into projections.

[1] Types with a binary operation with a unit.

The models of the sketch of unital magmas in Set are the unital magmas.

The models of the sketch of unital magmas in Set are the unital magmas.
Takeaways:

$$\mathrm{Mod}(\mathcal{S}, \mathrm{Mod}(\mathcal{T}, C)) \simeq \mathrm{Mod}(\mathcal{S} \otimes \mathcal{T}, C)$$

The models of the sketch of unital magmas in Set are the unital magmas.
Takeaways:

$$\mathrm{Mod}(\mathcal{S}, \mathrm{Mod}(\mathcal{T}, C)) \simeq \mathrm{Mod}(\mathcal{S} \otimes \mathcal{T}, C)$$

$$\boxed{\text{Models are functors/morphisms}}$$

The models of the sketch of unital magmas in Set are the unital magmas.
Takeaways:

$$\mathrm{Mod}(\mathcal{S}, \mathrm{Mod}(\mathcal{T}, C)) \simeq \mathrm{Mod}(\mathcal{S} \otimes \mathcal{T}, C)$$

Models are functors/morphisms

Model-taking is monoidally closed

An MMT-like theory
as a formal tool

A theory is a **sequence of declarations**. A declaration is **an identifier** and **an expression** built from previous declarations and constructors.

A theory is a **sequence of declarations**. A declaration is **an identifier** and **an expression** built from previous declarations and constructors.

For example:

theory **Monoid**

$$
\begin{aligned}
M &: \mathrm{Type} \\
\cdot &: M \to M \to M \\
e &: M \\
\mathsf{assoc} &: (x, y, z : M) \to x \cdot (y \cdot z) = (x \cdot y) \cdot z \\
\mathsf{unit}_{\mathrm{L}} &: (x : M) \to x = e \cdot x \\
\mathsf{unit}_{\mathrm{R}} &: x \cdot e = x
\end{aligned}
$$

With constructors $\to$, Type, dependent functions, and application

A morphism $\Sigma \to \Gamma$ takes every declared name in Σ to an expression in Γ in such a way as to preserve types under translation.

A morphism $\Sigma \to \Gamma$ takes every declared name in Σ to an expression in Γ in such a way as to preserve types under translation.

For example, a morphism **Monoid** $\to$ **Set**, for a set theory Set

$$M : \mathrm{Type} \mapsto X^X : \mathrm{Type} \quad (X \text{ some set})$$

$$\cdot : M \to M \to M \mapsto \lambda f \lambda g \lambda x \,.\, f(g(x)) : X^X \to X^X \to X^X$$

$$e : M \mapsto \lambda x \,.\, x : X^X$$

$$\mathsf{assoc} : \prod_{(x,y,z:M)} x \cdot (y \cdot z) = (x \cdot y) \cdot z \mapsto [\ldots] : [\ldots] \quad (\text{proof of associativity})$$

$$\mathsf{unit}_{\mathrm{L}} : (x : M) \to x = e \cdot x \mapsto [\ldots] : [\ldots] \quad (\text{proof of left unitality})$$

$$\mathsf{unit}_{\mathrm{R}} : x \cdot e = x \mapsto [\ldots] : [\ldots] \quad (\text{proof of right unitality})$$

No definition, but an example:

theory **Monoid**

M : Type

$\cdot : M \to M \to M$

$e : M$

assoc : $(x, y, z : M) \to x \cdot (y \cdot z) = (x \cdot y) \cdot z$

$\text{unit}_L : (x : M) \to x = e \cdot x$

$\text{unit}_R : x \cdot e = x$

$\lhook\joinrel\downarrow$

theory **Group**

$(\_)^{-1} : M \to M$

inverse : $(x : M) \to x \cdot x^{-1} = e$

theory **Sets**

$\vdots$

(X some set)

$\vdots$

$\downarrow$

theory **Something**

$(\_)^{-1} : X^X \to X^X$

inverse : $(f : X^X) \to f \circ f^{-1} = \mathrm{Id}_X$

MMT implements UL?

- Identification: definition, isomorphisms.
- Translation: morphisms.
- Combination: push-forwards, finite colimits in general.

The joys of modular mathematics

Grp

$$
\begin{aligned}
\mathsf{pt} &: \mathrm{Type} \\
\cdot &: \mathsf{pt} \to \mathsf{pt} \to \mathsf{pt} \\
\mathrm{e} &: \mathsf{pt} \\
(\ )^{-1} &: \mathsf{pt} \to \mathsf{pt} \\
\mathsf{assoc} &: (x, y, z : \mathsf{pt}) \to x \cdot (y \cdot z) = (x \cdot y) \cdot z \\
\mathsf{neutral}_{\mathrm{L}} &: (x : \mathsf{pt}) \to \mathrm{e} \cdot x = x \\
\mathsf{neutral}_{\mathrm{R}} &: (x : \mathsf{pt}) \to x = x \cdot \mathrm{e} \\
\mathsf{inverse} &: (x : \mathsf{pt}) \to x \cdot x^{-1} = \mathrm{e}
\end{aligned}
$$

Top*

$$
\begin{aligned}
\mathsf{pt} &: \mathrm{Type} \\
\mathsf{open} &: \mathcal{P}(\mathcal{P}(\mathsf{pt})) \\
\mathsf{base} &: \mathsf{pt} \\
\mathsf{open}_{\top} &: \mathsf{open}(\mathsf{full}_{\mathsf{pt}}) \\
\mathsf{open}_{\perp} &: \mathsf{open}(\mathsf{empty}_{\mathsf{pt}}) \\
\mathsf{open}_{\bigcup} &: (a : \mathrm{Type}) \to \\
&\quad (u : a \to \mathcal{P}(\mathsf{pt})) \to \\
&\quad ((i : a) \to \mathsf{open}(u_i)) \to \\
&\quad \mathsf{open}\left(\bigcup_i u_i\right) \\
\mathsf{open}_{\cap} &: (u, v : \mathcal{P}(\mathsf{pt})) \to \mathsf{open}(u) \to \mathsf{open}(v) \to \mathsf{open}(u \cap v)
\end{aligned}
$$

$$
\begin{aligned}
&\text{Cat} \\
\text{obj} &: \text{Type} \\
\text{arr} &: \text{obj} \to \text{obj} \to \text{Type} \\
\text{Id} &: (x : \text{obj}) \to \text{arr}\ x\ x \\
\circ &: (x, y, z : \text{obj}) \to \text{arr}\ y\ z \to \text{arr}\ x\ y \to \text{arr}\ x\ z \\
\mathsf{assoc} &: (x, y, z, w : \text{obj}) \to (f : \text{arr}\ x\ y) \to (g : \text{arr}\ y\ z) \to (h : \text{arr}\ z\ w) \to f \circ (g \circ h) = (f \circ g) \circ h \\
\text{Id}_{\text{L}} &: (x, y : \text{obj}) \to (f : \text{arr}\ x\ y) \to \text{Id}_x \circ f = f \\
\text{Id}_{\text{R}} &: (x, y : \text{obj}) \to (f : \text{arr}\ x\ y) \to f \circ \text{Id}_y = f
\end{aligned}
$$

Given a theory $(a_i : E_i)_i$, and given name C, its collectivization is

$$(a_i : E_i)_i \mapsto C : \text{Type}, (a_i : (c : C) \to \varphi(E_i))$$

where

$$\varphi(b) := b(c) \qquad b \text{ is an identifier}$$
$$\varphi(K(F_j; (x_{jk})_k)_j) := K(\varphi(F_j); (x_{jk})_k)_j$$

For example, for $\mathcal{C}(\mathrm{Grp})$ we have

$$
\begin{aligned}
&\text{Grp} \\
\\
\mathsf{pt} &: \text{Type} \\
\cdot &: \mathsf{pt} \to \mathsf{pt} \to \mathsf{pt} \\
\mathrm{e} &: \mathsf{pt} \\
(\ )^{-1} &: \mathsf{pt} \to \mathsf{pt} \\
\mathsf{assoc} &: (x, y, z : \mathsf{pt}) \to x \cdot (y \cdot z) = (x \cdot y) \cdot z \\
\mathsf{neutral}_{\mathrm{L}} &: (x : \mathsf{pt}) \to \mathrm{e} \cdot x = x \\
\mathsf{neutral}_{\mathrm{R}} &: (x : \mathsf{pt}) \to x = x \cdot \mathrm{e} \\
\mathsf{inverse} &: (x : \mathsf{pt}) \to x \cdot x^{-1} = \mathrm{e}
\end{aligned}
$$

$$
\begin{aligned}
&\mathcal{C}(\mathrm{Grp}) \\
\mathrm{Grp} &: \text{Type} \\
\mathsf{pt} &: (g : \mathrm{Grp}) \to \text{Type} \\
\cdot &: (g : \mathrm{Grp}) \to \mathsf{pt}_g \to \mathsf{pt}_g \to \mathsf{pt}_g \\
\mathrm{e} &: (g : \mathrm{Grp}) \to \mathsf{pt}_g \\
(\ )^{-1} &: (g : \mathrm{Grp}) \to \mathsf{pt}_g \to \mathsf{pt}_g \\
\mathsf{assoc} &: (g : \mathrm{Grp} \to (x, y, z : \mathsf{pt}_g) \to x \cdot_g (y \cdot_g z) = (x \cdot_g y) \cdot_g z \\
\mathsf{neutral}_{\mathrm{L}} &: (g : \mathrm{Grp}) \to (x : \mathsf{pt}_g) \to \mathrm{e}_g \cdot_g x = x \\
\mathsf{neutral}_{\mathrm{R}} &: (g : \mathrm{Grp}) \to (x : \mathsf{pt}_g) \to x = x \cdot_g \mathrm{e}_g \\
\mathsf{inverse} &: (g : \mathrm{Grp}) \to (x : \mathsf{pt}_g) \to x \cdot_g x^{-1_g} = \mathrm{e}_g
\end{aligned}
$$

Define the category of groups by interpreting Cat in $\mathcal{C}(\mathrm{Grp})$ (syn. by implementing the Cat interface in $\mathcal{C}(\mathrm{Grp})$.

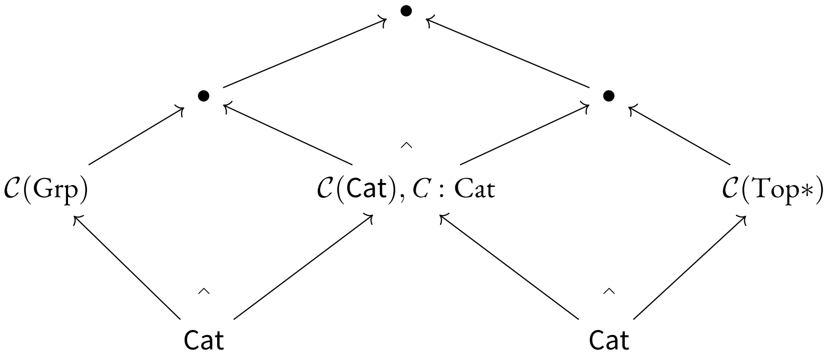
$$\begin{aligned}
\mathbf{Grp} : \mathrm{Cat} &\longrightarrow \mathcal{C}(\mathrm{Grp}) \\
\mathrm{obj} &\longmapsto \mathrm{Grp} \\
\mathrm{arr}\ G\ H &\longmapsto \sum_{f:\mathsf{pt}_G \to \mathsf{pt}_H} \prod_{x,y:\mathsf{pt}_G} (f(x \cdot_G y) = f(x) \cdot_H f(y),\ f(e_G) = e_H) \\
g \circ f &\longmapsto \langle \mathsf{fst}\ g \circ \mathsf{fst}\ f, \lambda x \lambda y\ .\ ? \rangle \\
\mathrm{Id}_G &\longmapsto \langle \mathrm{Id}_{\mathsf{pt}_G}, \lambda x \lambda y\ .\ \mathrm{Refl} \rangle \\
\mathsf{assoc} &\longmapsto \text{proof that } \circ \text{ is associative} \\
\mathrm{Id}_{\mathrm{L}} &\longmapsto \text{proof of left identity} \\
\mathrm{Id}_{\mathrm{R}} &\longmapsto \text{proof of right identity}
\end{aligned}$$

Do the same for Top*

To implement π_1, at least three options

1. Internal functor
2. Interpreted functor
3. Translation

Internal functor



$\mathcal{C}$(Grp)
$\mathcal{C}$(Cat), C : Cat
$\mathcal{C}$(Top∗)
Cat
Cat

Then,

$$\begin{aligned}
&\texttt{In}\ \mathbf{Grp} + \mathbf{Top}^* \\
\mathrm{obj}_{\pi_1} &: \mathrm{obj}(\mathbf{Top}^*) \to \mathrm{obj}((\mathit{Grp})) \\
\mathrm{obj}_{\pi_1} &:= \lambda X \left(\sum_{\gamma:\mathrm{arr}(I,X)} \gamma(0) = \gamma(1) = \mathsf{base}_X \right)
\end{aligned}$$

(Observe that $\mathrm{obj}(\mathbf{Top}^*)$ reduces to Top)

$$\begin{aligned}
\pi_1 &: \mathrm{Functor}(\mathbf{Top}, \mathbf{Grp}) \\
\pi_1 &:= \langle \mathrm{obj}_{\pi_1}, ?, ? \rangle
\end{aligned}$$

Interpreted functor

$$\begin{aligned}
&\text{Functor } \texttt{over } \mathcal{C}(\text{Cat}) \\
\mathsf{src} &: \text{Cat} \\
\mathsf{tgt} &: \text{Cat} \\
\mathsf{fobj} &: \text{obj}(\mathsf{src}) \to \text{obj}(\mathsf{tgt}) \\
\mathsf{map} &: (x, y : \text{obj}_{\mathsf{src}}) \to \text{arr}_{\mathsf{src}}(x, y) \to \text{arr}_{\mathsf{trg}}(\mathsf{fobj}(x), \mathsf{fobj}(y)) \\
\mathsf{coherence} &: (x, y, z : \text{obj}_{\mathsf{src}}) \\
&\to (f : \text{arr}_{\mathsf{src}}(x, y)) \\
&\to (g : \text{arr}_{\mathsf{src}}(y, z)) \\
&\to \mathsf{map}(g \circ_{\mathsf{src}} f) = \mathsf{map}(g) \circ_{\mathsf{trg}} \mathsf{map}(f)
\end{aligned}$$

$$
\begin{aligned}
\text{Functor} &\to \mathbf{Grp} + \mathbf{Top}^* \\
\mathsf{src} &\mapsto \mathbf{Top}^* \\
\mathsf{trg} &\mapsto \mathbf{Grp} \\
\mathsf{fobj} &\mapsto \lambda X \,.\, \pi_1(X) \\
\mathsf{map} &\mapsto \lambda f \,.\, [\textit{corresponding morphism}] \\
\mathsf{coherence} &\mapsto [\textit{proof of functoriality}]
\end{aligned}
$$

($\mathbf{Grp} + \mathbf{Top}^*$ is the co-product of Grp and Top* together with the vocabulary from Cat given interpretation under the morphisms **Grp** and $\mathbf{Top}^*$, respectively. That is realized as a suitable colimit.)

Translation

A morphism $\pi_1 : \mathrm{Grp} \to (\mathcal{C}(\mathrm{Top}^*), X : \mathrm{Top}^*)$:

$$
\begin{aligned}
\mathrm{Grp} &\to \mathcal{C}(\mathrm{Top}^*), C : \mathrm{Top} \\
\mathsf{pt} &\mapsto \prod_{\gamma : X^I} (\gamma(0) = \mathsf{base}_X, \gamma(1) = \mathsf{base}_X) \big/ \sim \\
\cdot &\mapsto \lambda\gamma, \delta, t \,.\, [\ldots] \\
\mathrm{e} &\mapsto [\![\mathsf{const}_{\mathsf{base}_X}]\!] \\
(\ )^{-1} &\mapsto [\ldots] \\
\mathsf{assoc} &\mapsto \text{proof of associativity} \\
\mathsf{neutral}_{\mathrm{L}} &\mapsto \text{proof of left neutrality} \\
\mathsf{neutral}_{\mathrm{R}} &\mapsto \text{proof of right neutrality} \\
\mathsf{inverse} &\mapsto \text{proof of inverseness}
\end{aligned}
$$

- A framework for modular mathematics making possible to
  - Do synthetic reasoning
  - In the shape of Universal Logic
- Missing: categorical structure and justification to theories

Bauer, Andrej (2006). “First Steps in Synthetic Computability Theory”. In: *Electronic Notes in Theoretical Computer Science* 155. DOI: 10.1016/j.entcs.2005.11.049.

Kock, Anders (2017). *New methods for old spaces: synthetic differential geometry*. arXiv. DOI: 10.48550/ARXIV.1610.00286. URL: https://arxiv.org/abs/1610.00286.

Mossakowski, Till et al. (2007). “What is a Logic?” In: *Logica Universalis*. Birkhäuser Basel. DOI: 10.1007/978-3-7643-8354-1_7.

nLab (2022). *Sketch*. URL: https://ncatlab.org/nlab/show/sketch (visited on 04/25/2022).