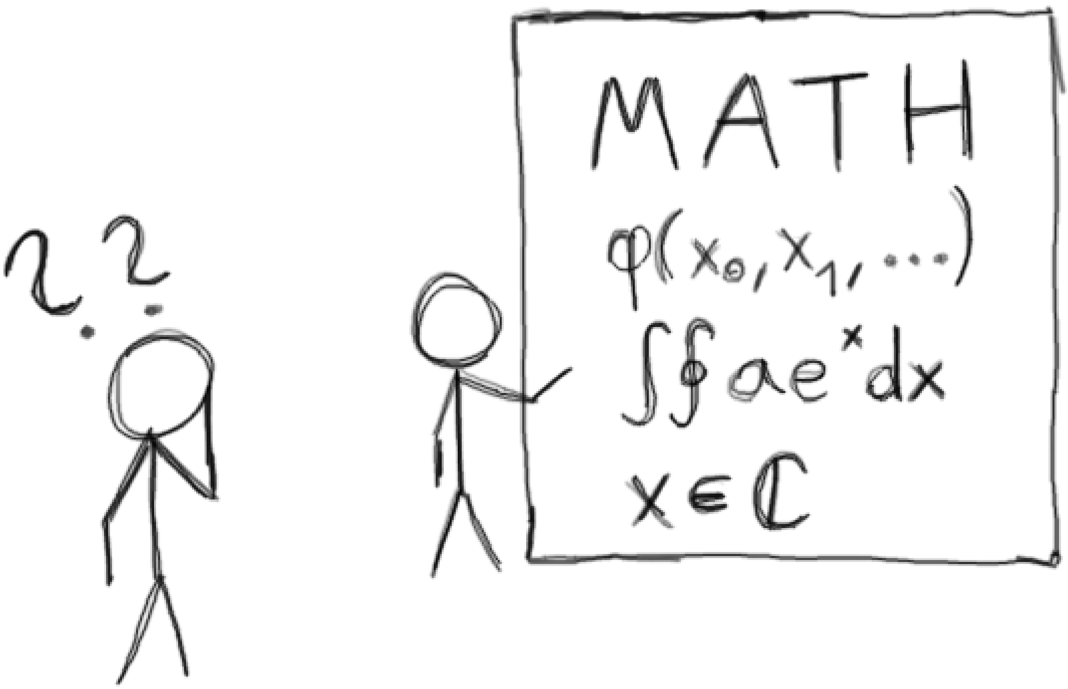
The History of Path Semantics

ILLUSTRATED
by Sven Nilsen, 2017

If you are like me, then you probably find a lot of math symbols[1] confusing.
What are they all about?

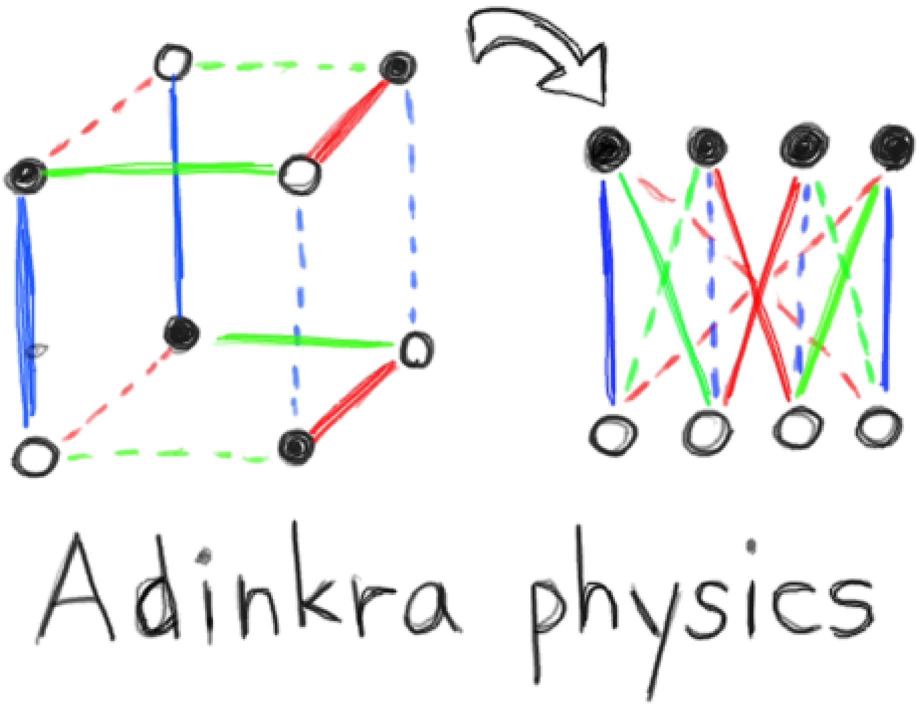


I have been fascinated by mathematics as long as I can remember, but the way it is presented makes it sometimes unappealing. The deeper ideas of mathematics[2] are often explained in a very abstract and theoretical way that most people do not understand.

Learning what all the symbols mean and how to use them can be a great challenge. It is like trying to dig through a mountain of knowledge to find what you are looking for.

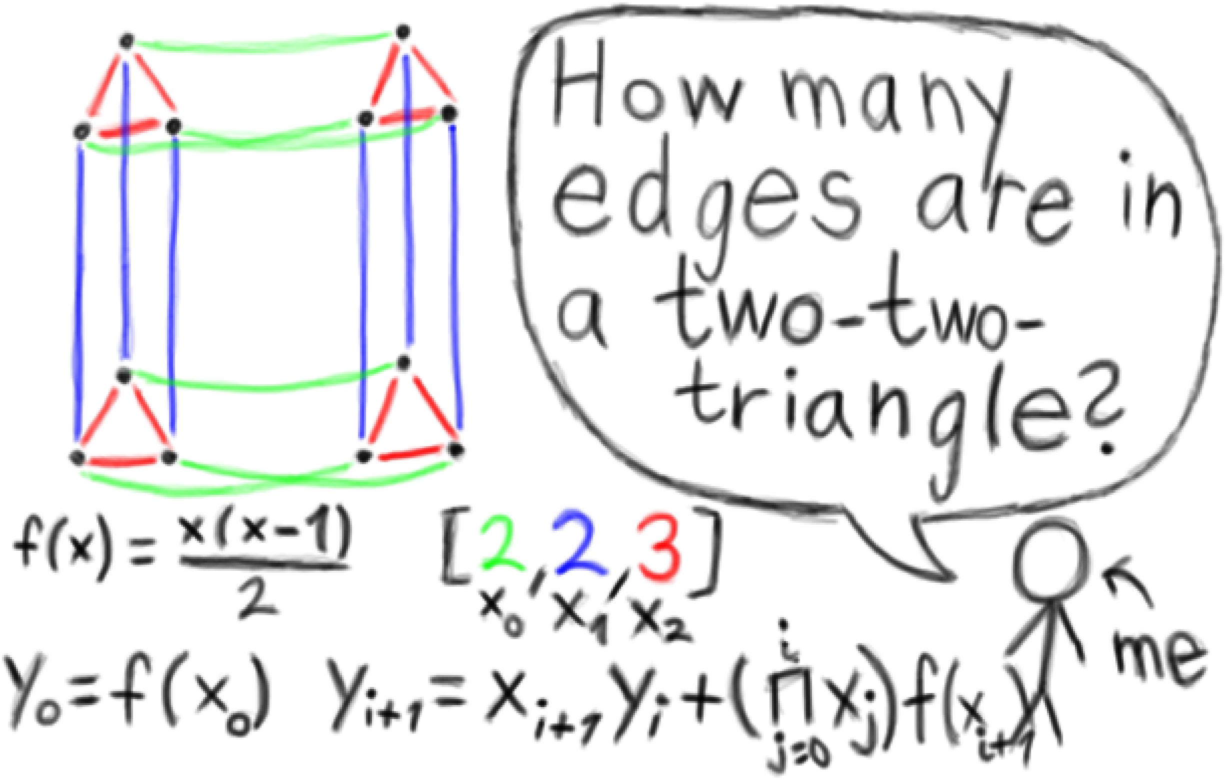
I think there should be a way to get to the core of what mathematics is about in a much shorter time. Path semantics[3] is my attempt to put advanced math at your fingertips with a very easy to understand notation.

My first idea of how to simplify mathematics came when I learned about Adinkra diagrams[4] that are used in theoretical physics.



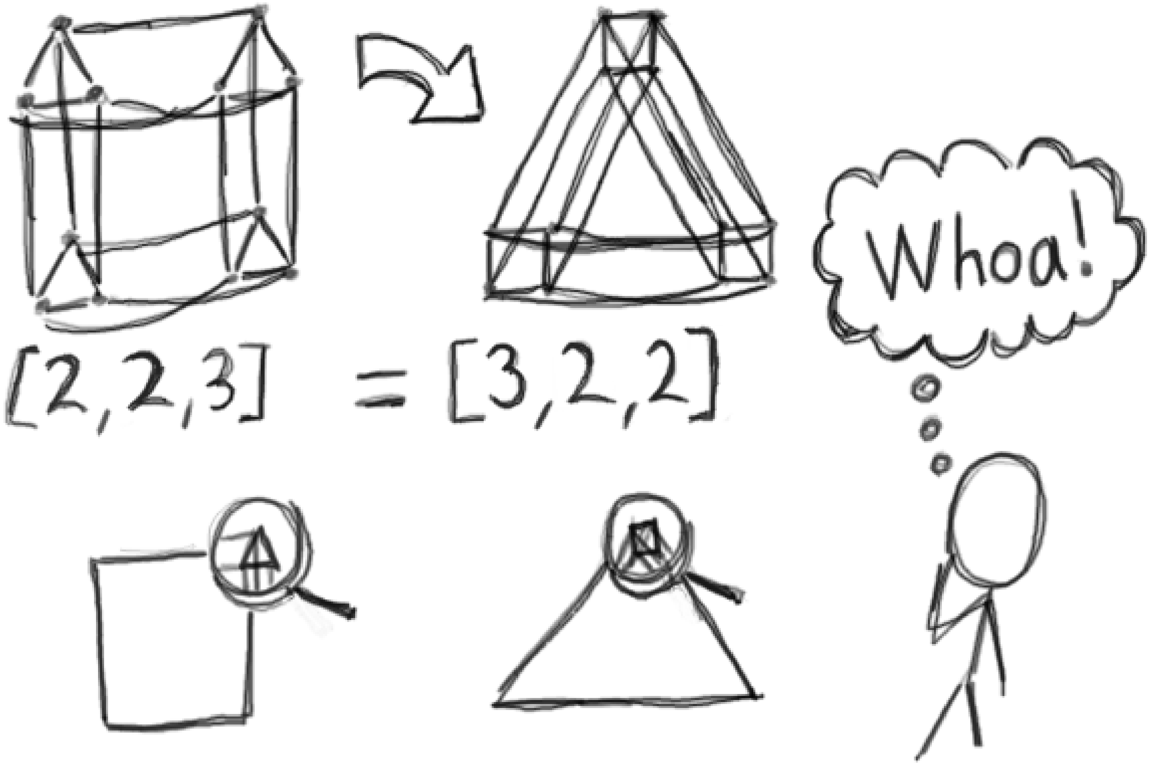
These diagrams can be transformed to describe equations in various ways. The manipulating of the diagram corresponds to mathematical operations on the equations.

I thought that if these diagrams could be generalized a bit, then they could describe more things.



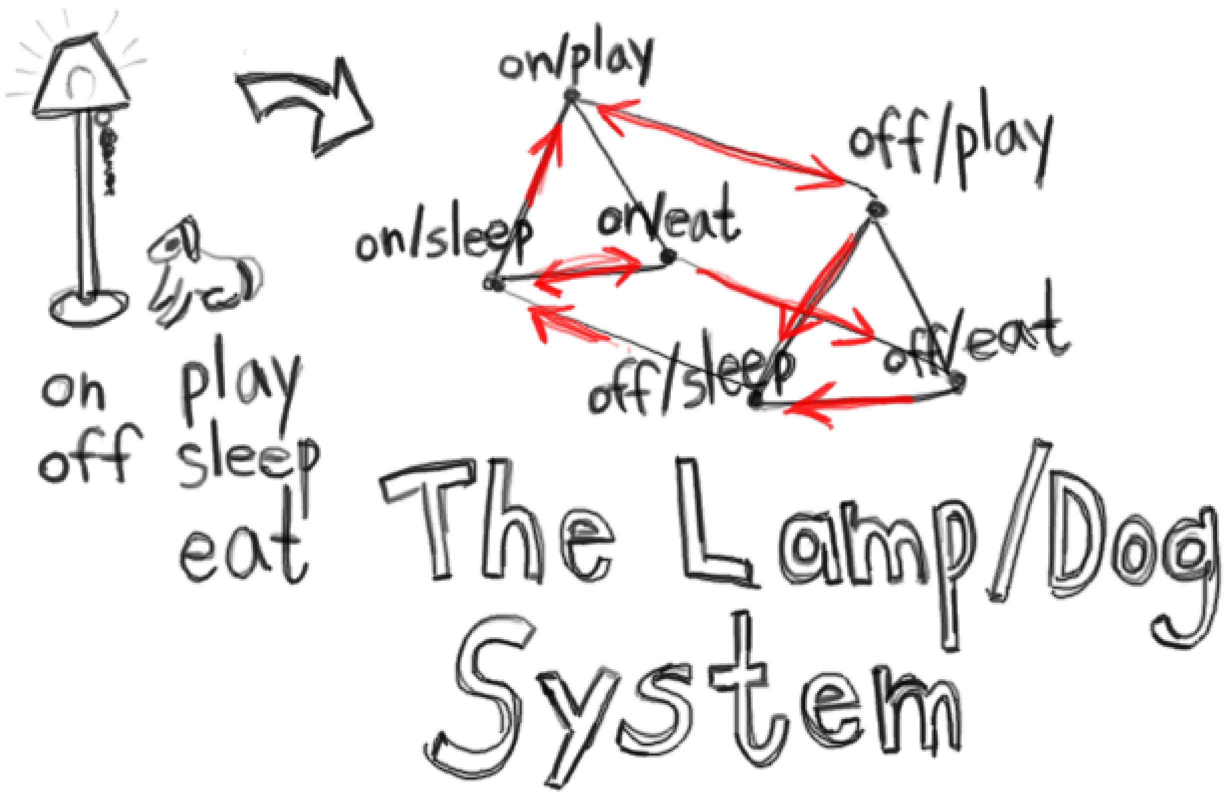
It took a while to figure out how it should look like and how to write the algorithms. Eventually, I got it done. In my “discrete” Rust library[5], the object is called `Context`[6] and `DirectedContext`[7].

One thing I discovered was that you could draw the same diagram in different ways and get two diagrams that looked differently, but actually they were equal.



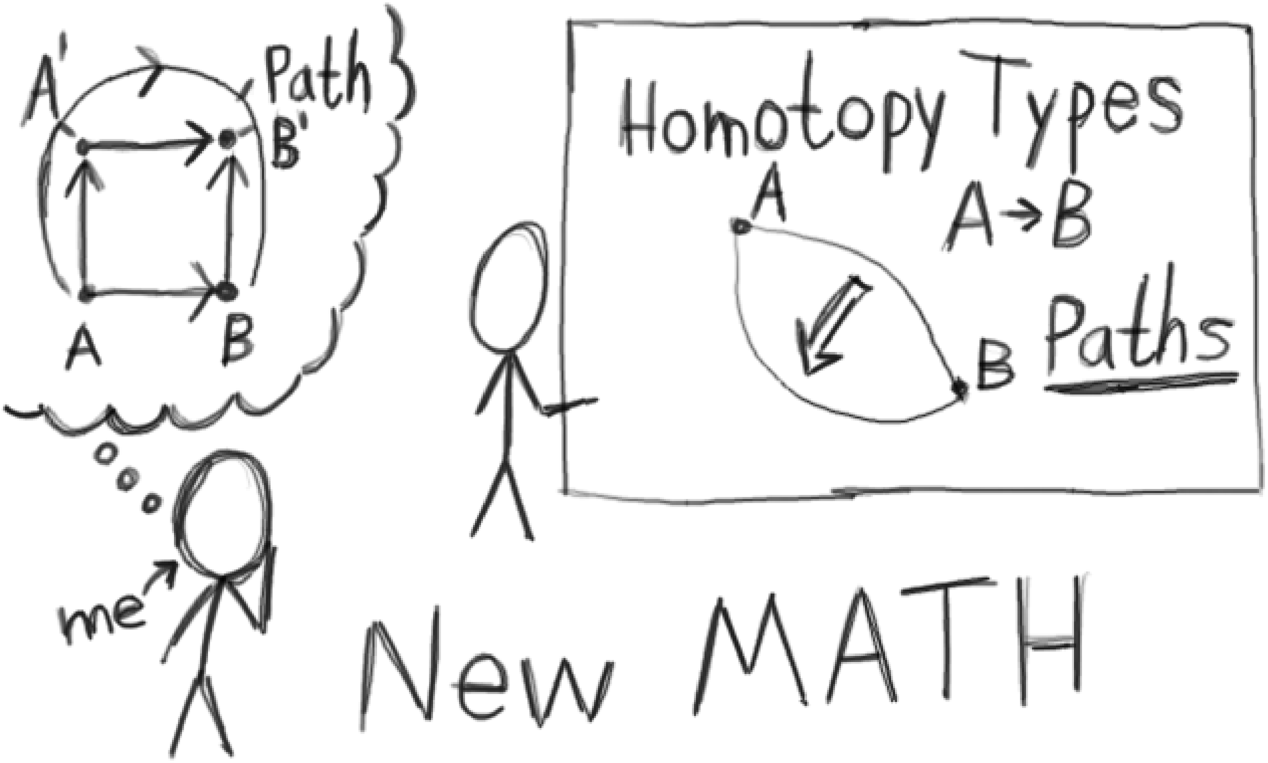
I imagined how any shape could secretly hide another shape in the corner. You never know for certain whether a rectangle is not hiding triangles and vice versa.

The cool thing about these diagrams is that they have one edge for every single change.



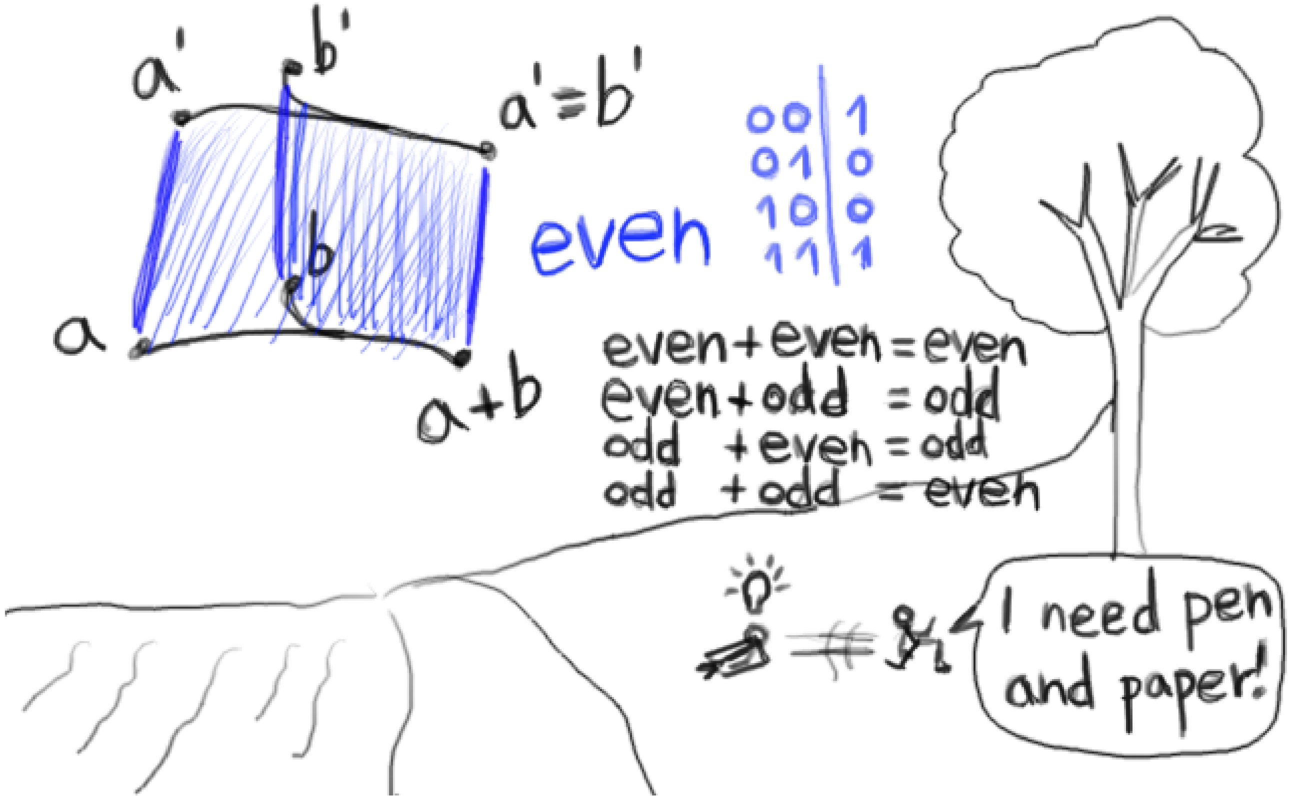
You can take any system that has a finite number of states and combine it with other systems to get a bigger one. The information encoded in the edges of a such diagram is how the larger system behaves. This is information that you can not find by studying the parts individually.

A new breakthrough in mathematics[8] connected dependently type theory with a field called "homotopy"[9]. The way I visualized it was different, but I started to think using the word "path".



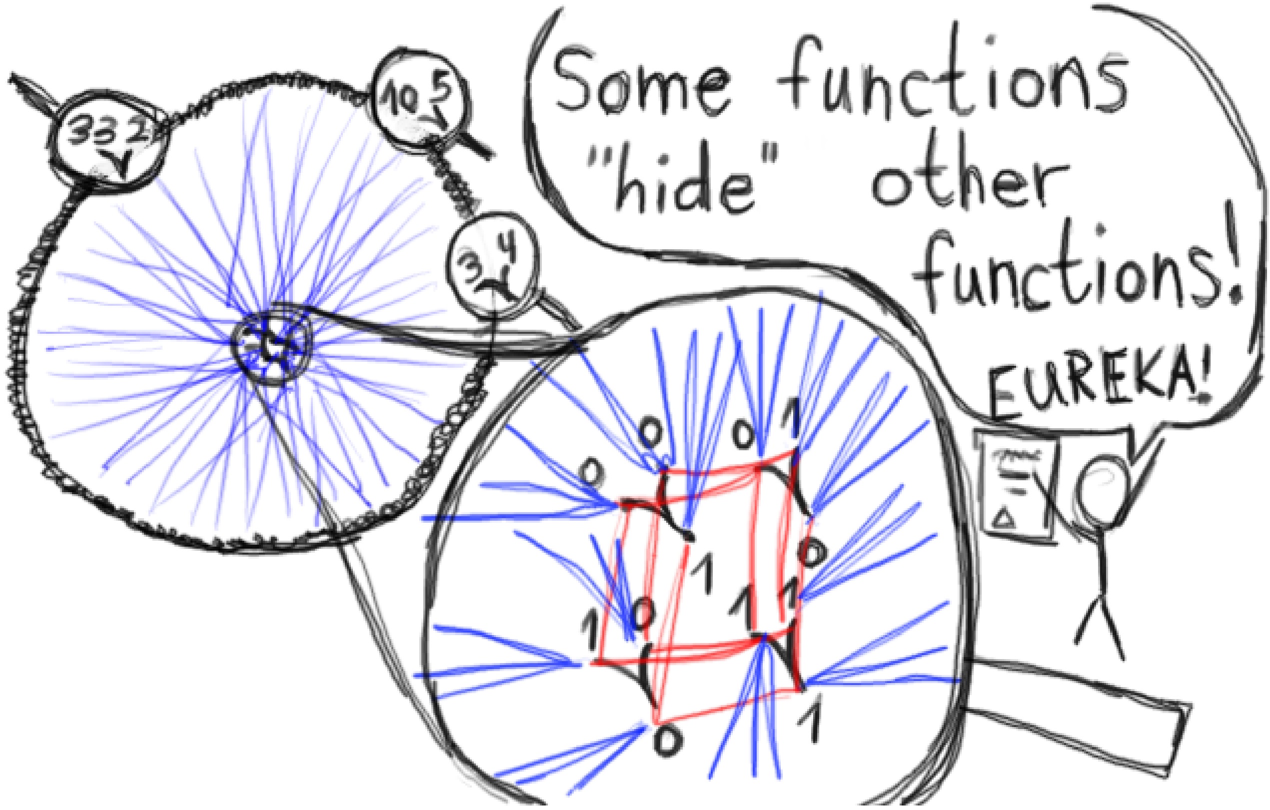
There could be more than one way to connect two objects. It can be described as paths to move around within a space where the end points are fixed. If you get to the same position, then you get to the same objects.

One day I was at the beach in the summer 2014, I thought about how adding and multiplying even numbers looked like the patterns of boolean functions. I got the idea they could be connected like in a space I used to describe with the diagrams.



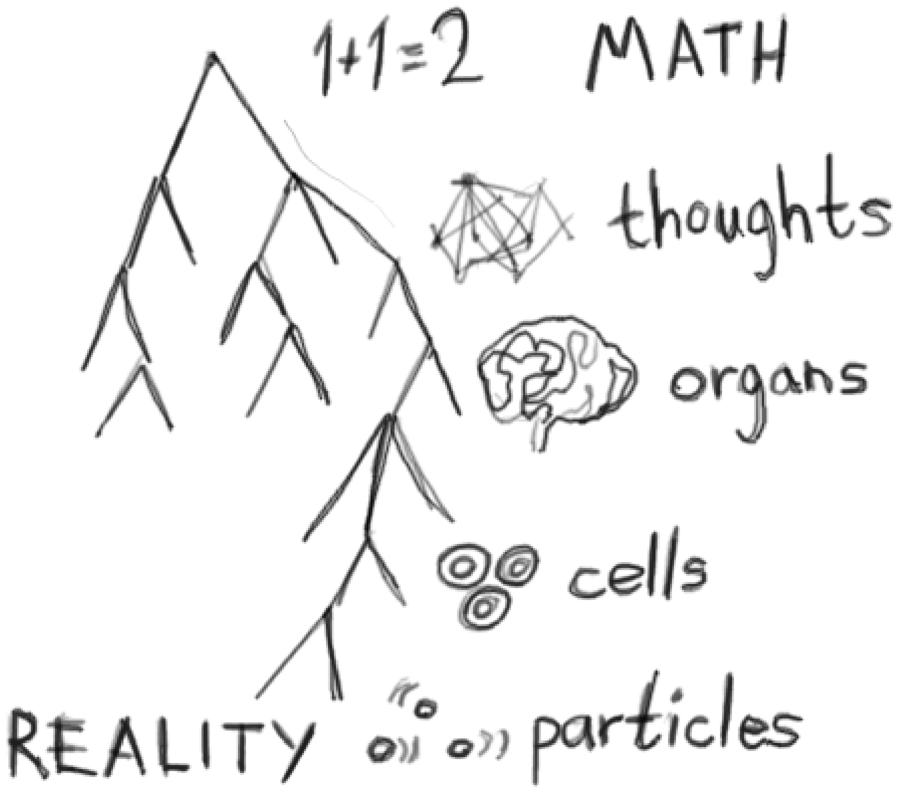
I ran home to write it down, and this was how path semantics started.

The core insight I had was that some functions “hide” other functions.



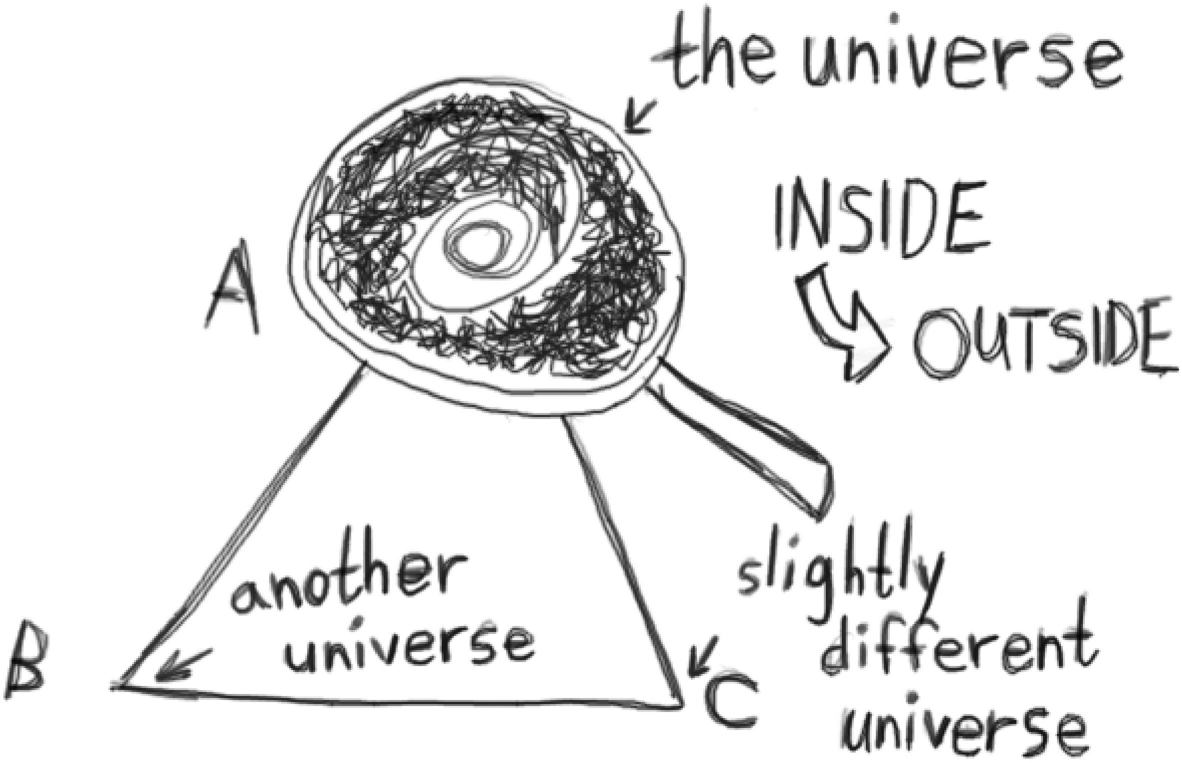
There can be infinitely many possible inputs to a function, but they all can lead to a tiny pattern by moving along a path in an abstract space that secretly connects functions.

Not only is this true for mathematical functions, but it is true for reality as well in the same way.



When we think of “1 + 1 = 2”, it is hidden within our thoughts, which is hidden inside the brain, which is hidden in a vast collection of cells, which is hidden in a soup of particles. If this was not the case, we could not think “1 + 1 = 2”. It is a strange thing that we are aware of what we are thinking.

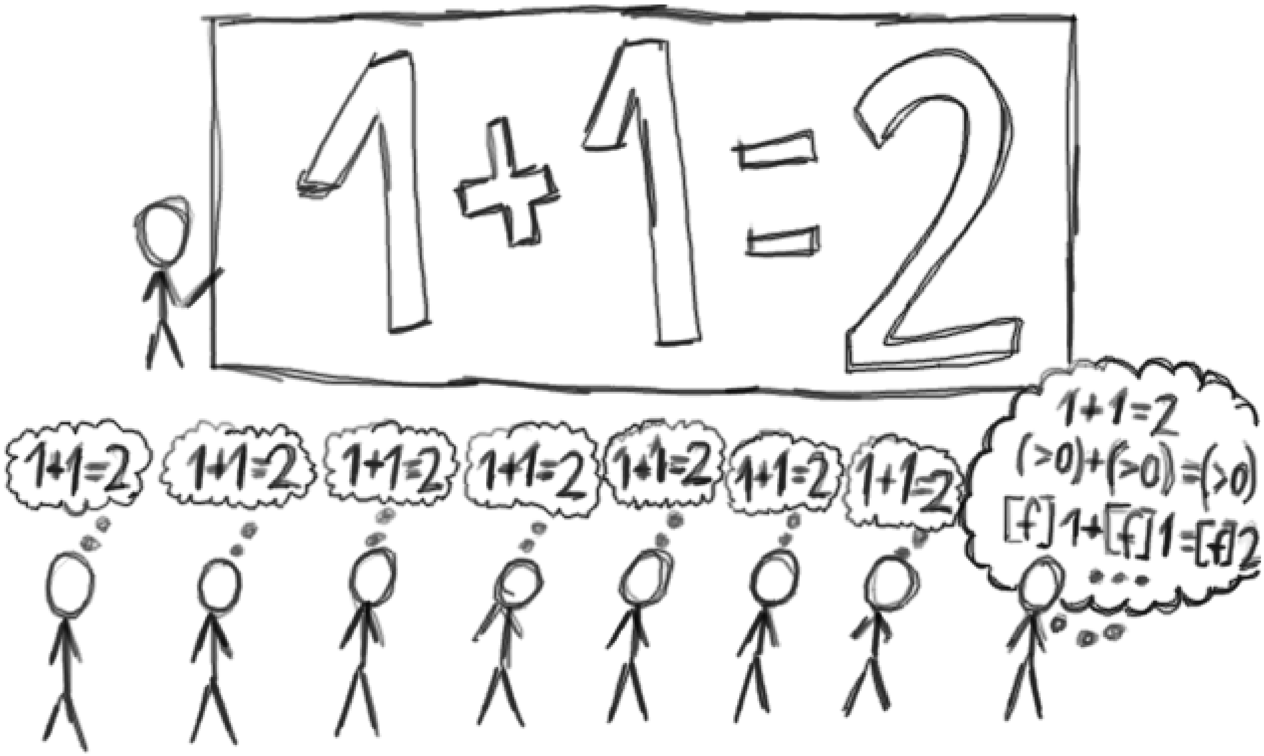
Just like the diagrams that can look very differently depending on how you draw them, perhaps the whole universe is hidden inside our thoughts? Perhaps not literally, but it could be what is hiddenly assumed when we talk about “truth” and “meaning”.



When we are thinking, it is like we are moving the whole universe, but how it moves depends on the way our thoughts are connected to the universe. There are laws for everything, so how we can move the universe is limited in part by our thinking and in part of how we exist.

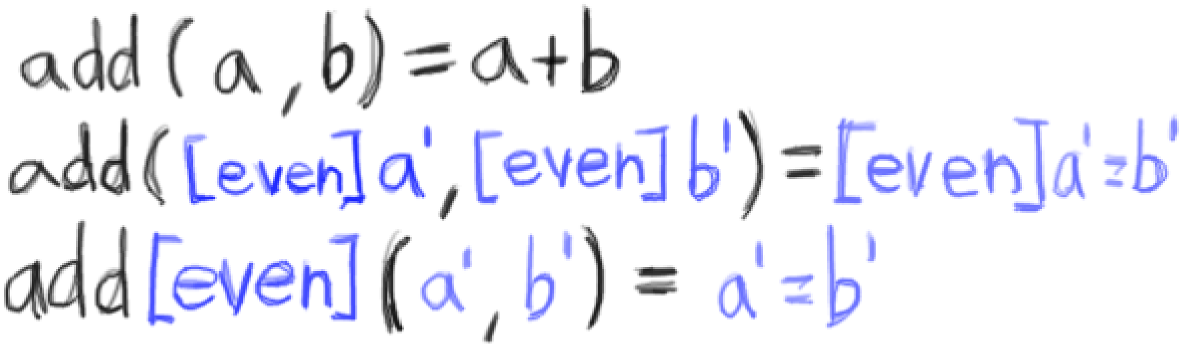
I want you to think that hidden thoughts inside a brain is equal to a hidden universe inside the thoughts. You can find out what people are thinking by scanning their brains, and you can “put back the universe” in your thoughts by transforming them with path semantics.

Most people are used to think of symbols as abstract representations only. This is very boring.



I believe the symbols hide secrets by simply existing in a real world where functions can be constructed. It is the existence of these functions that give meaning to what symbols can hide. Therefore, you are free to choose what to think about such abstractions, but only according to what others functions exist and whether there is some input that gives the value.

To describe what symbols hide, I use other functions put in a square bracket `[]`.



The square brackets can be factored out if they are the same, and this becomes a symmetric path[10].

Much later I found a way to write asymmetric paths in the same way, so now the notation is much more powerful and feels more complete.

Here is an example of asymmetric path notation[11]:

$$f[g_0 \times g_1 \rightarrow g_2] <=> h$$

References:

[1] “List of mathematical symbols”
Wikipedia
https://en.wikipedia.org/wiki/List_of_mathematical_symbols

[2] “Category Theory”
Stanford Encyclopedia of Philosophy
https://plato.stanford.edu/entries/category-theory/

[3] “Path semantics”
AdvancedResearch, Sven Nilsen
https://github.com/advancedresearch/path_semantics

[4] “Adinkra symbols (physics)”
Wikipedia
https://en.wikipedia.org/wiki/Adinkra_symbols_(physics)

[5] “Discrete – combinatorial phantom types for discrete mathematics”
AdvancedResearch, Sven Nilsen
https://github.com/advancedresearch/discrete

[6] “Context structure (discrete combinatorial mathematics)”
AdvancedResearch, Sven Nilsen
https://docs.rs/discrete/0.2.0/discrete/struct.Context.html

[7] “DirectedContext structure (discrete combinatorial mathematics)”
AdvancedResearch, Sven Nilsen
https://docs.rs/discrete/0.2.0/discrete/struct.DirectedContext.html

[8] Homotopy Type Theory
https://homotopytypetheory.org/

[9] “Homotopy”
Wikipedia
https://en.wikipedia.org/wiki/Homotopy

[10] “Path Semantics Cheat Sheet”
Sven Nilsen, 2017
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/path-semantics-cheat-sheet.pdf

[11] “Algebraic notation for asymmetric paths”
Sven Nilsen, 2017
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/algebraic-notation-for-asymmetric-paths.pdf