Modeling Complex Systems with Category Theory

Daniel Sinderson

2024

Southern Oregon University

Objective and Scope of the Project

Scope of the Project

My objective for this project was to learn about the brand new field of categorical systems theory , which applies the pure math field of category theory to the study of arbitrary systems , and then use the tools that I learned to model a complex real-world system.

But I didn't know category theory or any complex systems...

The Actual Scope of the Project

1. Learn Category Theory
2. Learn Categorical Systems Theory
3. Write a software package for it in Python
4. Learn about a real world system (Transcription Networks)
5. Model it and simulate the results
6. Live to tell you all about it

Scope of this Presentation

Keep things high-level and descriptive and move fast.

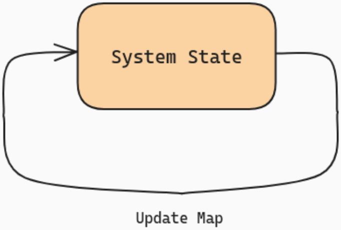
1. Tell you what a system is.
2. Tell you what a category is.
3. Tell you how they can work together.
4. Show you a complex system and how I modeled it.
5. Talk about the simulation results.

Systems

What is a System?

A system is a thing that changes. At it's simplest, a system consists of the following two things.

1. A set of states that the system can be in.
2. A map that updates the system's state based on the state that it's currently in.



Examples of Systems

The set of natural numbers with the successor function is a system.

$$\begin{aligned} S : \mathbb{N} &\to \mathbb{N} \\ n &\mapsto n+1 \end{aligned}$$

Examples of Systems

An ordinary differential equation is also a system.

$$\frac{dx}{dt} = \kappa x$$

Here the set of states is the set of real numbers $\mathbb{R}$ and the update map is the differential equation itself: given the current state, x, the differential equation tells us how to change it.

What’s the Problem

The problem with these systems is that they’re closed.

They don’t interact with each other. And they don’t interact with their environment.

So let’s open them up with category theory.

Categories

Category Theory Basics

Category theory is a theory of structure and composition.

It uses the mathematical object of a category to encapsulate the notion of associative composition.

With this alone it creates a language capable of formalizing all of mathematics.

What is a Category?

Definition (Category)
A category $\mathcal{C}$ is defined by the following:

1. $\mathcal{C}$ contains a collection of objects $\text{ob}(\mathcal{C})$.
2. For any two objects $a, b \in \mathcal{C}$ there is a collection of morphisms, $f : a \to b$, between those objects called the homset, $\mathcal{C}(a, b)$.
3. Every object $a \in \mathcal{C}$ has a morphism to itself $\text{id}_a : a \to a$ called its identity.
4. For every two morphisms $f : a \to b$ and $g : b \to c$ there's a third morphism $g \circ f : a \to c$ that's their composition.

These objects and morphisms are then under two constraints: unitality and associativity.

Definition (Category cont. Unitality)
Any morphism $f : a \to b$ can be composed with the identity morphisms of a and b such that $f \circ \mathrm{id}_a = \mathrm{id}_b \circ f = f$.

$$\begin{tikzcd} a \arrow[r, "f"] \arrow[d, "\mathrm{id}_a"'] & b \arrow[d, "\mathrm{id}_b"] \\ a \arrow[r, "f"'] & b \end{tikzcd}$$

What is a Category?

These objects and morphisms are then under two constraints: unitality and associativity.

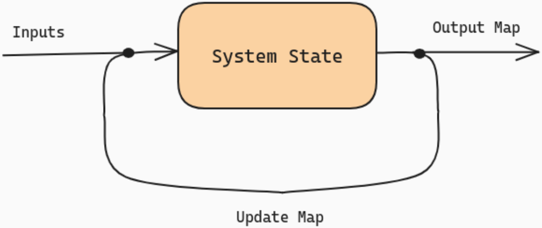
Definition (Category cont. Associativity)
For any morphisms $f : a \to b$, $g : b \to c$, and $h : c \to d$, $h \circ (g \circ f) = (h \circ g) \circ f$. Since it doesn't matter what order we apply the morphisms, we write this $h \circ g \circ f$.

$$
\begin{tikzcd}
a \arrow[r, "f"'] \arrow[rr, bend left, "g\circ f"] & b \arrow[r, "g"'] \arrow[rr, bend right, "h\circ g"'] & c \arrow[r, "h"'] & d
\end{tikzcd}
$$

Composing Systems

What is a Open System?

An open system is as a system with an interface.



Lenses and Arenas

To model this mathematically we'll use objects called arenas and morphisms called lenses.

Definition (Arenas)

An arena is a pair of objects $\begin{pmatrix} A \\ B \end{pmatrix}$ from an underlying category.

Definition

A lens is a morphism between arenas made from a pair of morphisms from the underlying category

$$\begin{pmatrix} f \\ g \end{pmatrix} : \begin{pmatrix} A \\ B \end{pmatrix} \leftrightarrows \begin{pmatrix} C \\ D \end{pmatrix}$$

where $f : B \to D$ is the output map and $g : B \times C \to A$ is the update map.

What this Gives Us

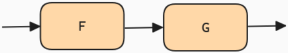
Arenas and lenses between them form a monoidal category.

This means that we can compose systems and multiply them.

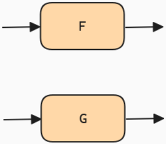
By modeling our open systems as lenses, composition of lenses lets us place systems in series and multiplication of lenses lets us place systems in parallel.

Systems in Series and Parallel

Systems in Series: $\begin{pmatrix} g^{\#} \\ g \end{pmatrix} \circ \begin{pmatrix} f^{\#} \\ f \end{pmatrix}$

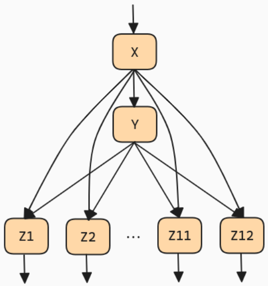


Systems in Parallel: $\begin{pmatrix} g^{\#} \\ g \end{pmatrix} \otimes \begin{pmatrix} f^{\#} \\ f \end{pmatrix}$

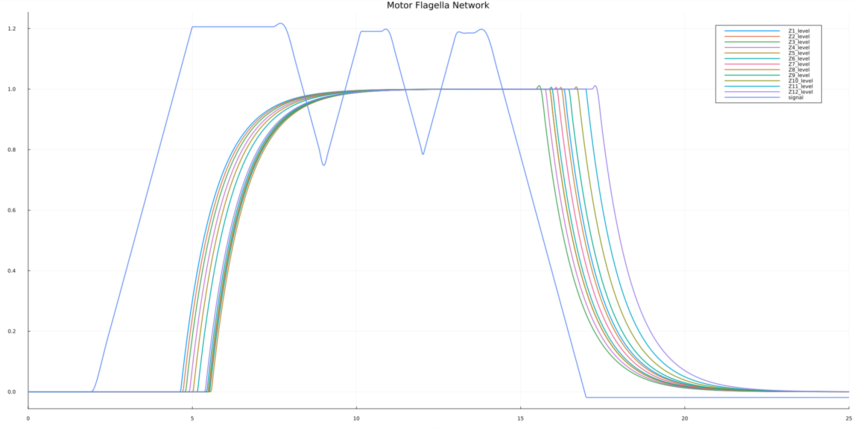


Case Study

Motor Flagella Network



Motor Flagella Network Simulation



Results

Conclusion