COMPOSITIONAL MODELING WITH STOCK-FLOW DIAGRAMS

John Baez, Xiaoyan Li, Nathaniel Osgood, Sophie Libkind and Evan Patterson

ACT 2022
There is a community of epidemiologists who use “stock-flow diagrams” to model the spread of disease. This includes Nathaniel Osgood and Xiaoyan Li, who do COVID modeling for the Canadian government.
There is a systematic procedure to turn stock-flow diagrams into dynamical systems — that is, systems of differential equations. This is how stock-flow diagrams are most often used for modeling.
There is a systematic procedure to turn stock-flow diagrams into dynamical systems — that is, systems of differential equations. This is how stock-flow diagrams are most often used for modeling.

If we build a large stock-flow diagram by composing smaller \textit{open} stock-flow diagrams, its dynamical system is the composite of \textit{open} dynamical systems for these smaller open diagrams. This is \textbf{compositional modeling}.
There is a systematic procedure to turn stock-flow diagrams into dynamical systems — that is, systems of differential equations. This is how stock-flow diagrams are most often used for modeling.

If we build a large stock-flow diagram by composing smaller open stock-flow diagrams, its dynamical system is the composite of open dynamical systems for these smaller open diagrams. This is compositional modeling.

Unfortunately, modeling with stock-flow diagrams is done using software that does not support compositional modeling — most commonly AnyLogic.
Here is Osgood and Li’s COVID model used by the government of Canada, in AnyLogic:
Applied category theory to the rescue!

Together with Evan Patterson, Sophie Libkin and myself, Osgood and Li have now created StockFlow: software that supports compositional modeling with stock-flow diagrams.

We used AlgebraicJulia: a framework for high-performance scientific computing that lets you program using category theory. This is being developed by a team including James Fairbanks, Evan Patterson, Sophie Libkind and many others.

Let me sketch the math underlying our software. I’ll only do a simplified version of stock-flow diagrams.
In its simplest form, a **stock-flow diagram** consists of finite sets and functions:

\[
\begin{align*}
\text{Flows} & \xrightarrow{u} \text{Stocks} \\
\text{Flows} & \xleftarrow{d} \text{Stocks} \\
\text{Links} & \xrightarrow{t} \text{Stocks} \\
\text{Links} & \xleftarrow{s} \text{Stocks}
\end{align*}
\]

together with, for each \( f \in \text{Flows} \), a function \( \phi_f : \mathbb{R}^{L(f)} \to \mathbb{R} \) where \( L(f) \) is the set of links whose target is \( f \).
Define a **dynamical system** on a finite set $N$ to be a continuous vector field $\mathbf{v}$ on $\mathbb{R}^N$. This gives a differential equation

$$\frac{d}{dt}x(t) = \mathbf{v}(x(t))$$

describing how the stocks $x(t) \in \mathbb{R}^N$ change with time.

Each stock-flow diagram with set $N$ of stocks gives a dynamical system on $N$:

\[
\begin{align*}
\frac{dS}{dt} &= -\phi_i(S, I) \\
\frac{dl}{dt} &= \phi_i(S, I) - \phi_r(I) - \phi_d(I) \\
\frac{dR}{dt} &= \phi_r(I) \\
\frac{dD}{dt} &= \phi_d(I)
\end{align*}
\]
Using the theory of decorated cospans, there is a category \( \text{Open}(\text{StockFlow}) \) where objects are finite sets and morphisms are \emph{open} stock-flow diagrams:
Using the theory of decorated cospans, there is a category $\text{Open}(\text{StockFlow})$ where objects are finite sets and morphisms are open stock-flow diagrams:

(Well, really isomorphism classes of open stock-flow diagrams.)
There is also a category Open(Dynam) where objects are finite sets and morphisms are open dynamical systems:

\[ A \xrightarrow{i} N \xleftarrow{o} B \quad \nu \in \text{Vect}(N) \]

where \( \text{Vect}(N) \) is the set of continuous vector fields on \( \mathbb{R}^N \).

(Again, we really need isomorphism classes of open dynamical systems.)
There is a functor

\[ \nu : \text{Open} (\text{StockFlow}) \rightarrow \text{Open} (\text{Dynam}) \]

sending each finite set to itself and each open stock-flow diagram to the corresponding open dynamical system.
By implementing some of these ideas in AlgebraicJulia, my coauthors created a software package called **StockFlow**, now available on GitHub. This lets you:

- compose open stock-flow diagrams
- turn open stock-flow diagrams into open dynamical systems
- *solve* the differential equations given by these dynamical systems.
StockFlow lets you compose not only decorated cospans, but also “multicospans”:
Osgood, Li and others are building a graphical interface for StockFlow. This allows teams to collaboratively build stock-flow models on their web browsers. Existing software is single-user, not web-based, and does not allow composition.
In August 2022, Osgood, Libkind, Patterson, Li, Fairbanks and others will teach a week-long course on StockFlow. This is being funded by the Canadian Network for Modelling Infectious Disease (CANMOD).

So, applied category theory can be a practical tool!

Here’s our paper: