Compositional Active Inference

A “Process Theory” for Finding Right Abstractions

Toby St Clere Smithe

Topos Institute
&
University of Oxford

12 May 2021

Some Background

My background is in computational neuroscience. (I'm a DPhil student at Oxford.)

But I was dissatisfied with its mathematical structure:

- How do models relate, or plug together?
- How can we translate ideas between groups?
- How can we stop reinventing concepts?

I realized I was seeking a compositional *lingua franca*,
sufficient to express common patterns, translate ideas, and connect things together ...

something like category theory, in fact ...

This Talk

So today I'm going to tell you some things I have learned,
about the "structure of learning structure".

I'm going to emphasize compositionality.
This entails taking interaction seriously.

I'm going to sketch the beginnings of a new story:

- How do adaptive systems find abstractions?
  - (How do things learn anything?)
- How are their beliefs structured?
  - (Do we ever have more than consensus?)

At Topos, we're interested in telling such new stories for all kinds of systems:
we seek a new, interconnected, scientific ontology.

We hope that acknowledging interaction promotes sustainability.

Overview

Compositionality at Large (1/3)

The world is complicated — and so are the things within it!

But at the same time, it seems to be made of parts plugged together.

So: if we are to have a hope of understanding complexity,
we might as well make use of this latent structure.

- Isn't "making use of latent structure" what "right abstractions" are all about?

Compositionality at Large (2/3)

Categorical and compositional structures are everywhere:

- Maps of the world
- Topological spaces
- Neural circuits
- Quantum circuits
- Bayesian networks
- Dynamical systems
- Logical statements
- Natural-language statements
- Recipes
- Computer programs

Especially, any kind of ‘open’ system ...

Compositionality at Large (3/3)

Taking connectedness and compositionality seriously
means acknowledging — *really* acknowledging! — two related things:

1. Interaction
2. Context-dependence

After all: only tautologies are true in a vacuous context, without assumptions!

Theorem (Fundamental Theorem of Category Theory)

Yoneda's Lemma *says: the ways that some thing connects to others determine the thing itself.*

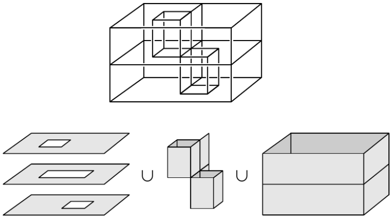
Firth's maxim: "You shall know a word [or place or concept or ...] by the company it keeps".

This applies equally to our abstractions, our models, and the stories they tell.

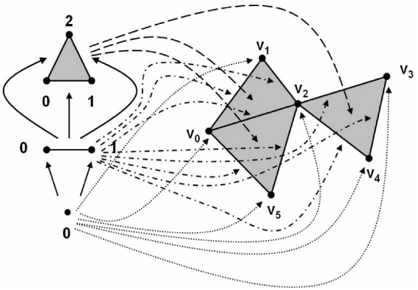
Compositional Spaces

Space is the stage on which events occur: where we exist, and over which our beliefs lie.

Space itself is compositional, by 'gluing':



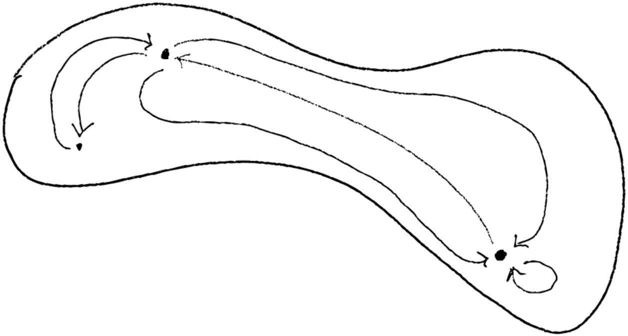
Hatcher's 'house' [1]



A simplicial complex [2]

Categorified Spaces

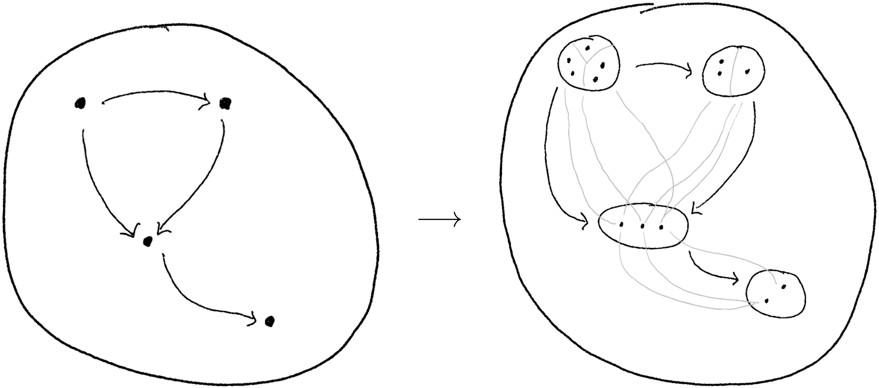
Spaces are more than just collections of points glued together:
there are also *paths* between the points.



We can 'categorify' spaces accordingly: paths become morphisms.
This points to *topos theory*.

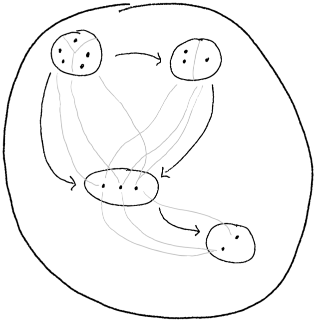
Topoi as Spaces

A topos is a particularly nice kind of categorified space.
In a "big topos", we can think of the points (objects) themselves as spaces, and the paths as maps between spaces:

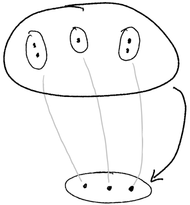


Bundles and Dependent Types

The *slice* over an object X is the category of objects with maps into X.
We can think of this as: “the topos **in the context of** X”.



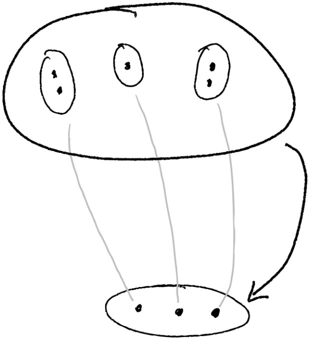
Its objects are *bundles*:



Bundles represent *dependent types*:
“types with context”.

Think: “places” and “stuff that can happen in each place”.
Examples: weather forecast; any sheaf (attaching data to spaces); polynomial functors ...

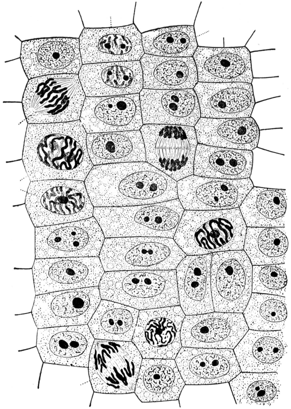
Logic and Dependence

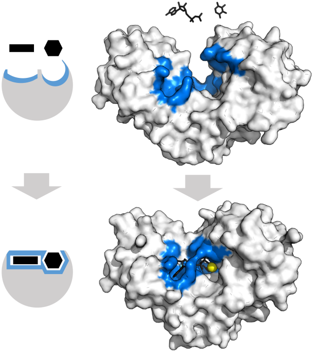


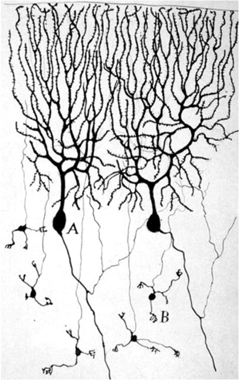
- Each topos comes with a powerful "internal language":
  - a logic over its corresponding space.
- This logic is higher-order:
  - We have conjunctions, disjunctions, implications; and quantifiers.
- We have a "subobject classifier", assigning truth-values to propositions:
  - Roughly, 'where' a proposition is true.
- And we can construct dependent types, too.
  - The slice over an object B is the topos of dependent types over B.
  - B gives us a logical context:
    - it encapsulates our axioms, or
    - our assumptions about the structure of the space.
  - Changing our assumptions (our context) corresponds to moving around the base topos (where B lives).

Compositional Things

The world is full of things, not just spaces!
Biology, full of interconnection, supplies numerous examples:





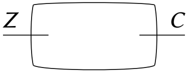


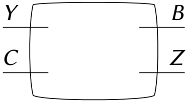
(pictures from Wikipedia)

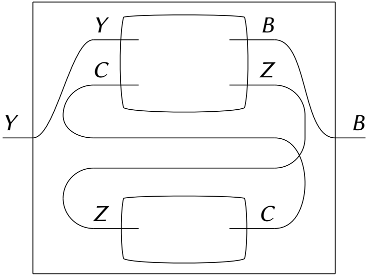
And of course: artificial objects, machines, etc. We can formalize their interconnection.

Polynomials for Interconnection

We can describe the interconnection of systems using *polynomials* [3]: (input to output :: left to right)







Cy^Z

BZy^{YC}

$BZy^{YC} \otimes Cy^Z \to By^Y$

NB: 'Polynomial' notation: coefficients are outputs, exponents are inputs

We will see later that these shapes can be much more sophisticated.
But ... *how do we know what stuff is out there?*

Overview

Whence the élan vital?

So we can start to talk about stuff. But what about animated stuff?
And how does "animated stuff" itself know about stuff?

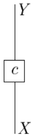
More precisely, we'll ask the following questions:

- What is the structure of an agent's internal model?
  - So: what is the structure of Bayesian inversion?
  - What structure underlies the "modularity of mind"?
- How do these models relate to the (interconnection of) their forms?

Introducing Markov Categories

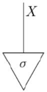
We work in a *Markov* or *copy-delete* category – **canonical example**: $\mathcal{K}\ell(\mathcal{D})$

Objects: spaces X, Y sets X, Y



Morphisms: "stochastic channels" *ie.* functions from points to 'beliefs'

$X \to\!\!\!\!\bullet\, Y$ $\qquad X \to \mathcal{D}Y \;\cong\; X \times Y \to [0,1]$



States: channels out of the monoidal unit *ie.* probability distributions (formal convex sums)

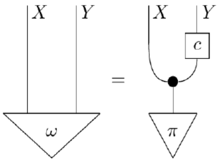
$I \to\!\!\!\!\bullet\, X$ $\qquad X \to [0,1] \qquad \sum_{x:X} \boxed{p(x)}\; |x\rangle$

so general channels are like 'conditional' probability distributions,
and we adopt the standard notation $p(y|x) := p(x)(y)$

Composition: given $p : X \to\!\!\!\!\bullet\, Y$ and $q : Y \to\!\!\!\!\bullet\, Z$, "average over" Y — for example:

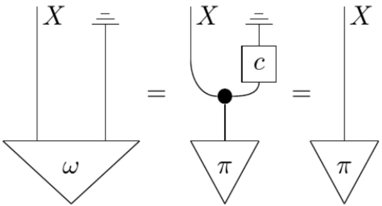
$$q \bullet p : X \to \mathcal{D}Z := x \mapsto \sum_{z:Z} \boxed{\sum_{y:Y} q(z|y) \cdot p(y|x)}\; |z\rangle$$

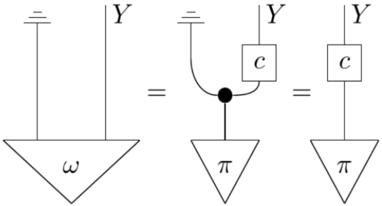
Joint states and generative models



$$\mathrm{P}_{\omega}(A, B) = \mathrm{P}_{c}(B|A) \cdot \mathrm{P}_{\pi}(A)$$

With two *marginals* given by discarding:

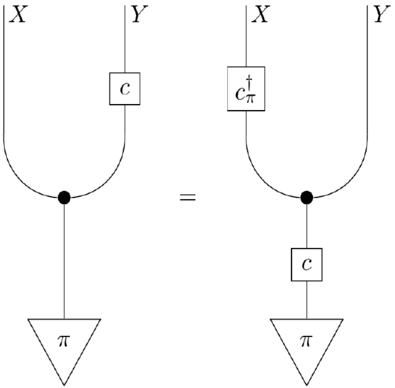




Bayes' rule, categorically

As a 'dagger' operation on stochastic channels:

$$\mathrm{P}_c(B|A)\cdot\mathrm{P}_\pi(A) = \mathrm{P}_{c^\dagger_\pi}(A|B)\cdot\mathrm{P}_{c\bullet\pi}(B)$$



Note that the Bayesian inverse channel depends on the prior!

The bidirectionality of Bayesian inference

- Given a 'generative' channel $c : X \to \mathcal{D}Y$, the corresponding 'recognition' channel has a **state-dependent** type $c^\dagger : \mathcal{D}X \times Y \to \mathcal{D}X$.
- A pair of a forwards map with a 'dependent' backwards map is called a **lens**.
- Lenses are a common pattern in 'bidirectional' systems (economic games, databases, etc).
  - They encapsulate what David Jaz was calling "hierarchical planners".
- The inverse of a composite channel (hierarchical generative model) is the lens composite of its components [4].
- This explains formally the bidirectional structure of predictive coding (cf. right).
  - A "right abstraction" !

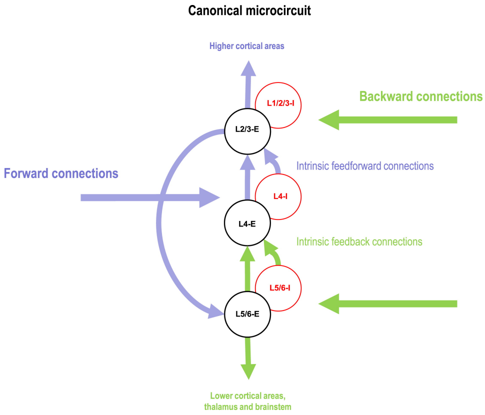
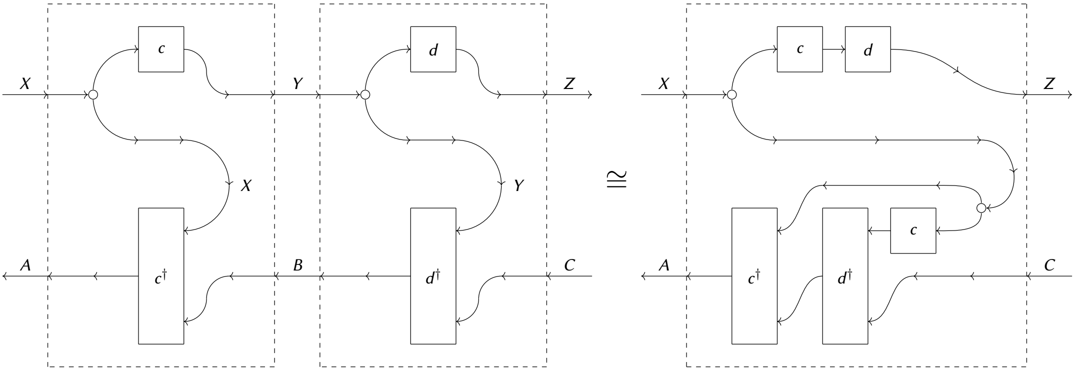


Figure: Bastos et al. [5]

Bayesian lenses

These *Bayesian lenses* form a category of morphisms between (pairs of) spaces.
These morphisms (the lenses) compose like this:



This captures the structure of inverting a hierarchical or causal model.
(Example: brain's visual predictions at cinema...)

Statistical Games for Approximate Inference

- Given a stochastic channel $c : X \to \mathcal{D}Y$ and a prior $\pi : \mathcal{D}X$, we can form the inversion $c^\dagger_\pi : Y \to \mathcal{D}X$. And we've seen that these pairs form lenses.
  Why is this useful?
- Typically, obtaining $c^\dagger_\pi$ is computationally difficult: we usually need to approximate it.
- This gives us a lot of freedom. Often, one approximation scheme might be 'better' than another, and we should like to quantify this.
- And, often, the fitness of our approximation depends on how it interacts with the world: the prior we choose, and the dataset we have.
- So the approximation is typically *context-dependent* and *parameterized*.

The Category of Statistical Games

The objects of **SGame** will be the objects (X, A) of **BayesLens**.

Then a **statistical game** is a morphism $(X, A) \to (Y, B)$ in **SGame**:
a lens $(X, A) \nrightarrow (Y, B)$ paired with a contextual *loss function* $\operatorname{ctx}\big((X, A), (Y, B)\big) \to \mathbb{R}$.

$\operatorname{ctx}\big((X, A), (Y, B)\big)$ is the set of contexts for lenses $(X, A) \nrightarrow (Y, B)$:

- Everything needed to "close off" the lens.
- $\operatorname{ctx}\big((X, A), (Y, B)\big) = \mathbf{BayesLens}\big((1, 1), (X, A)\big) \times \mathbf{BayesLens}\big((Y, B), (1, 1)\big)$
- That is: a prior $\pi : \mathcal{D}X$ on X and a *continuation* channel $Y \to \mathcal{D}B$.

Composition of statistical games is lens composition paired with the sum of the 'local' fitnesses.
Identities are identity lenses (which just pass on information) with 0 fitness.

Example: Maximum Likelihood Estimation

A Bayesian lens of the form $(1, 1) \nrightarrow (X, X)$ is fully specified by a state $\pi : 1 \bullet\!\!\rightarrow X$.

A context for such a lens is given by a trivial 'prior' on 1 and $k : X \bullet\!\!\rightarrow X$ is any endochannel on X. (Idea: "given a prediction, obtain a random observation".)

A **maximum likelihood game** is any game of type $(1, 1) \rightarrow (X, X)$ for any $X : \mathcal{C}$, and whose loss function is $\mathbb{E}_{k \bullet \pi}\left[-\log p_\pi\right]$, where p_π is a density function for π.

NB: $\mathbb{E}_{x \sim \pi}\left[f\right] = \mathbb{E}(f \bullet \pi) = \int_{x:X} f(x)\, \pi(\mathrm{d}x)$ when $f : X \rightarrow \mathbb{R}$ and $\pi : \mathcal{C}(1, X)$.

Thinking of density as a measure of likelihood, note that an optimal strategy for an ML game is one that maximises the likelihood of the state obtained from the context.

Example: Bayesian Inference

By generalizing the forwards part of the lens from states to channels, we obtain the following.

A **simple Bayesian inference game** is any game of type $(Z, Z) \to (X, X)$ with loss function

$$\mathop{\mathbb{E}}_{x \sim k \bullet c \bullet \pi} \left[D_{KL}(c'_\pi(x), c^\dagger_\pi(x)) \right]$$

where $\pi : I \to\!\!\!\!\!\bullet\; Z$, $k : X \to\!\!\!\!\!\bullet\; X$, D_{KL} denotes the Kullback-Leibler divergence,
and $c^\dagger$ is the exact inversion of c.

NB: since Bayesian updates compose optically, these games are closed under composition, giving hierarchical Bayesian inference games.

Example: Free Energy Game

Typically, computing $D_{KL}(c'_\pi(x), c^\dagger_\pi(x))$ is hard, so systems approximate.
A common approximation is the *free energy*, which bounds $D_{KL}(c'_\pi(x), c^\dagger_\pi(x))$ from above.
So, a **free energy game** is any game of type $(Z, Z) \to (X, X)$ with loss function

$$\begin{aligned}
&\mathop{\mathbb{E}}_{x \sim k \bullet c \bullet \pi}\left[\mathop{\mathbb{E}}_{z \sim c'_\pi(x)}[-\log p_c(x|z)] + D_{KL}(c'_\pi(x), \pi)\right] \\
&= \mathop{\mathbb{E}}_{z \sim c'_\pi \bullet k \bullet c \bullet \pi}\left[-\int_X \log p_c(\mathrm{d}k \bullet c \bullet \pi | z)\right] + D_{KL}(c'_\pi \bullet k \bullet c \bullet \pi, \pi)
\end{aligned}$$

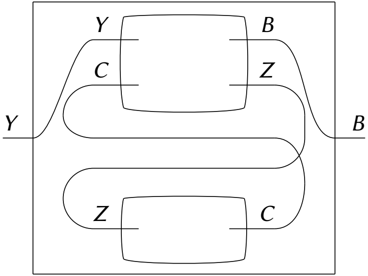
where $\pi : I \mathrel{\bullet\!\!\to} Z$ and $k : X \mathrel{\bullet\!\!\to} X$, and where the equality holds by linearity of expectation.
(It's easy to check that this gives an upper bound on $D_{KL}(c'_\pi(x), c^\dagger_\pi(x))$, so I won't..)

... But how do we breathe life into these systems? How do they *find abstractions*?

Overview

Polynomial Morphology (1/2)

Polynomials represent interfaces;
their morphisms patterns of interconnection:

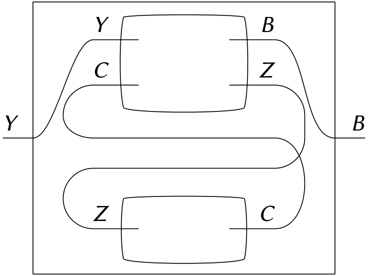


$$BZy^{YC} \otimes Cy^{Z} \to By^{Y}$$

But that's not all!

Polynomial Morphology (1/2)

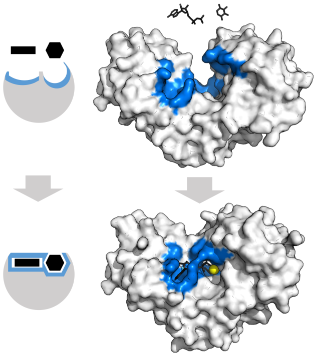
Polynomials represent interfaces;
their morphisms patterns of interconnection:



$$BZy^{YC} \otimes Cy^{Z} \to By^{Y}$$

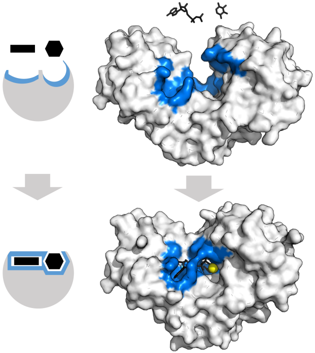
But that's not all!

Polynomials can describe much richer kinds of interconnection:



Even systems that change their shape!
How does this work?

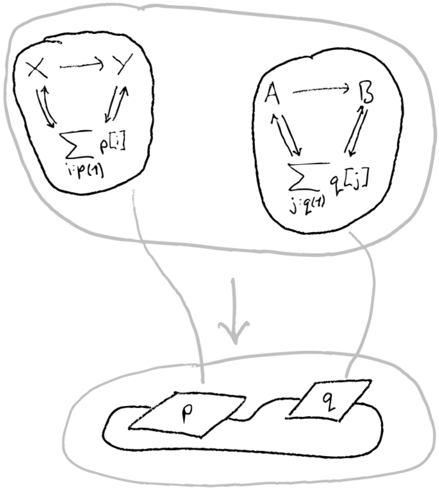
Polynomial Morphology (2/2)



- A general polynomial p looks like: $Ay^X + By^Y + Cy^Z + \cdots$
- More generally, that is: $\sum_{i:p(1)} p[i]$
  - For example: $p(1) = A + B + C$; $p[i]_{i \in A} = X$, etc
  - So $p : \sum_{i:p(1)} p[i] \to p(1)$ is a bundle over $p(1)$!
- I like to think of each polynomial as a **phenotype**:
  - The base type $p(1)$ is **configurations** or **morphology**;
  - Each $p[i]$ is the type of **immanent signals** or 'sensorium'.
  - Think hedgehog, or "Markov blanket"!
- Then poly. morphisms model interactions just as on the left!
- We can also assign positions to configurations using dependent polynomials.
  - The 'base' polynomial represents the shape of the 'world'.
  - (But we'll not get into these details now...)

Internal Models: Polynomially-Indexed Statistical Games

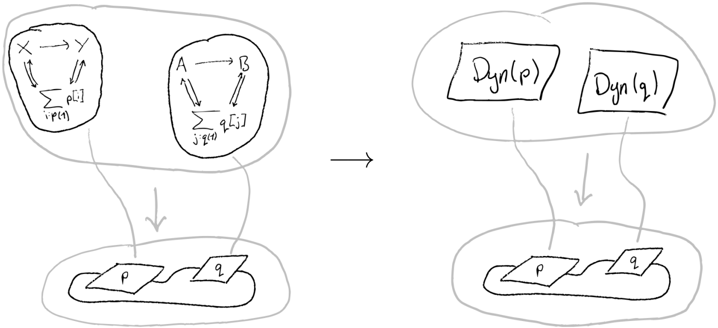
Now, we can start to animate these polynomial shapes.



- We can 'index' the category of statistical games by polynomials: like a slice category.
- That is, for each polynomial p, we obtain a category of "statistical games over p".
- This gives us a fibration: a categorified bundle (left).
  - Each object over p is a "generative model of p-sensations".
  - By sampling from the predicted configurations, we get something like 'action'.
    - (What more is action than a change in configuration?)
- We also get a recipe for describing the generative model of a composite ('multi-agent') system, in terms of the component systems' models.

Active Inference Doctrines

But statistical games themselves are not dynamical! Hence: **active inference doctrines**.
In the literature, there are many recipes for turning statistical games into dynamical systems.
Formally, these organize themselves into indexed functors:



The resulting dynamical systems **perceive** their sensoria, and change their configurations in **action**, to bring their beliefs into alignment with 'reality', and improve their chances of survival, or ability to find abstractions, or their fulfilment, ... or *whatever*.

Systems with Volition (1/2)

Recall that a system's performance at a statistical game is contextual:
it depends on prior beliefs (inc. its model structure!), and on its environment.

To improve its performance a system can:

1. Update its beliefs (perception);
2. update its model structure (part of its beliefs);
   - (NB. the 'latent' domain X of the internal model is roughly the "space of external states")
3. change its shape (a kind of action);
4. couple with part of the world, and change that shape (another kind of action).

By equipping it with 'strong' initial beliefs, we can give it preferences for particular world states.
And the system will attempt to achieve those states: *i.e.*, display **volition**.

Systems with Volition (2/2)

We can sketch a couple of examples, of increasing complexity:

Homeostasis

- Suppose: system's sensorium includes a key parameter, such as the ambient temperature.
- Suppose: by adjusting configuration, the system can move around to sample the parameter.
- Suppose: the 'prior' encodes a high-precision distribution centred on the acceptable range of this parameter.
- Then: by minimizing free energy, the system will attempt to configure itself to remain within the acceptable parameter range.

Systems with Volition (2/2)

We can sketch a couple of examples, of increasing complexity:

Homeostasis

- Suppose: system's sensorium includes a key parameter, such as the ambient temperature.
- Suppose: by adjusting configuration, the system can move around to sample the parameter.
- Suppose: the 'prior' encodes a high-precision distribution centred on the acceptable range of this parameter.
- Then: by minimizing free energy, the system will attempt to configure itself to remain within the acceptable parameter range.

Morphogenesis (very briefly...)

- Suppose: multiple 'homeostasis' systems, each sensing some signalling molecule.
- Suppose: poly map encodes pattern of signal molecule concentrations.
- Suppose: 'priors' encode "target local configurations".
- Then: free-energy minimization induces systems to arrange themselves to obtain global target pattern.

The "Internal Universe" of an Active System

- Speculatively, we can also model *navigation* in arbitrary spaces: we can use dependent polynomials to attach the system to positions in some world; and this world may in turn be a topos.
- Then: there need not be any restriction on the form of the internal model.
- In particular, it too might be structured like a topos:
  - Model parameters encode topos structure (the context, or "world shape").
  - So: updating model parameters means learning "what maps where", given the context.
- The system's state representation gives location (topos object).
- Action corresponds to following a morphism in the topos.
  - (Sense-data varies 'fibrationally'.)
- Putting a prior on a particular location induces system to reach that location, learning the structure along the way.
  - So: *navigating a space is just like finding a mathematical proof!*
  - Points towards 'well-typed' theory of cognition ...
- In principle: use internal language to put logical constraints on the target state?
- Lots left to figure out here ...

Concluding Remarks (1/2)

- It seems increasingly natural to think of the "world inside my head" like a topos.
  - So: structure-learning becomes logical "context-learning".
  - The brain assigns beliefs to propositions (a kind of classifying object).
  - Represent beliefs about processes using Cartesian closed structure (internal implication)?
- Then can ask: how do different systems' internal universes compare?
  - Do we have consensus? What are the obstructions?

- "Finding right abstractions" seems to mean: finding good explanations.
  - In our world, more often than not, these should be compositional; hence, reusable.
  - More poetically: finding compelling stories.
    - (The good ones hang around!)
- So: in a complex world, considering systems in isolation is unhelpful.
  - Always ask: "what's the context?"

Concluding Remarks (2/2)

Of course, there's lots of work still to do. Amongst other questions:

- To what extent does the process by which we (=our brains) find abstractions agree with 'black-boxing', and other accounts?
    - What's missing?
- How do these processes relate to causal / temporal inference?
- Can we see explicit signatures of "well-typed cognition" in the brain?
- Can we use the internal language of a topos to encode constraints on active systems' behaviours?
- How can we compare systems' internal universes?

References

[1] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002. 556 pp. ISBN: 0521795400. URL: http://pi.math.cornell.edu/~hatcher/AT/ATpage.html.

[2] Greg Friedman. "An elementary illustrated introduction to simplicial sets". In: *Rocky Mountain J. Math. 42 (2012), no. 2, 353-423* (09/24/2008). arXiv: 0809.4221v5 [math.AT].

[3] David I. Spivak. "Poly: An abundant categorical setting for mode-dependent dynamics". In: (05/05/2020). arXiv: 2005.01894 [math.CT].

[4] Toby St. Clere Smithe. "Bayesian Updates Compose Optically". In: (05/31/2020). arXiv: 2006.01631v1 [math.CT].

[5] A. M. Bastos et al. "Canonical microcircuits for predictive coding". In: *Neuron* 76.4 (11/2012), pp. 695–711. DOI: 10.1016/j.neuron.2012.10.038.