The role of embodiment in predictive coding frameworks

A mathematical perspective

Manuel Baltieri - 8th July 2024

ARAYA

Contents

"Using mathematical symbols =/= doing mathematics" (Sophie Libkind)

Goal: Mathematise aspects of predictive coding/FEP

Delivery: Long intro + short "punchline"

Contents

- Some connections to ASSC
- Agent-environment systems
- Predictive coding preliminaries and some "commandments"
- Coalgebras for dynamical systems by example (maps between systems and behaviours)
- Bisimulations to study embodiment in predictive coding proposals
- Some reflections on embodiment

A tale of two ideas

Structure and behaviour

- Mathematical structure of conscious experience

Structuralism is a theory of consciousness that seeks to analyze the elements of mental experiences, such as sensations, mental images, and feelings, and how these elements combine to form more complex experiences.

Structuralism was founded by Wilhelm Wundt, who used controlled methods, such as introspection,to break down consciousness to its basic elements without sacrificing any of the properties of the whole.

Lopez-Garrido 2023

- Computational functionalism

Computational functionalism is the view that computations of some kind are sufficient to instantiate consciousness. It derives from the popular philosophical notion of functionalism, which says – roughly – that consciousness is a matter only of what a system does, not of what it is made out of (Putnam, 1975).

Seth 2024

Duality of structure and behaviour

Algebras and coalgebras

Structure vs. Observation

Posted by Emily Riehl

guest post by ***Stelios Tsampas*** *and* ***Amin Karamlou***

Today we'll be talking about the theory of universal algebra, and its less well-known counterpart of universal coalgebra. We'll try to convince you that these two frameworks provide us with suitable tools for studying a fundamental duality that arises between *structure* and *behaviour*. Rather than jumping straight into the mathematical details we'll start with a few motivating examples that arise in the setting of functional programming. We'll talk more about the mathematics at play behind the scenes in the second half of this post.

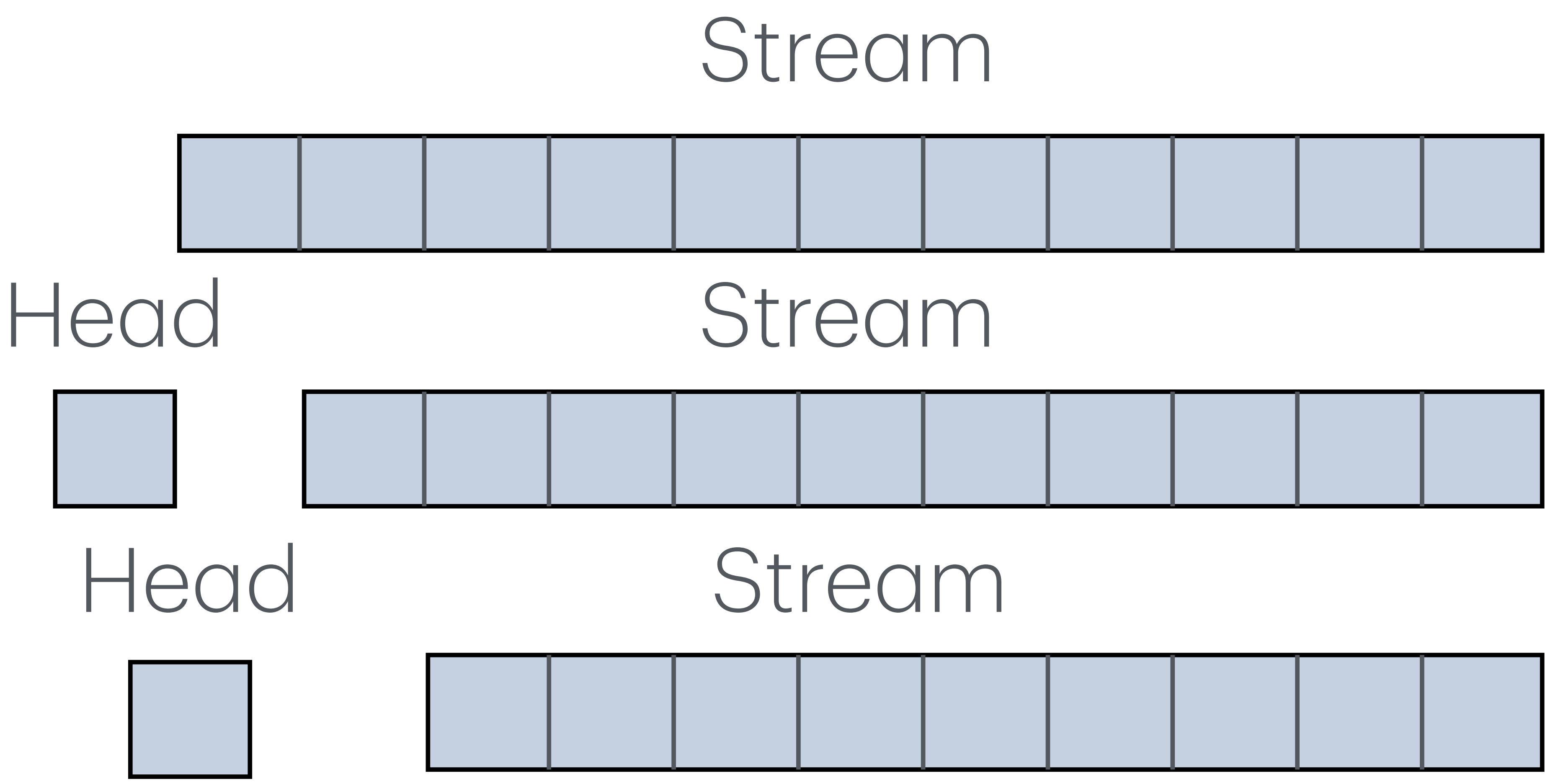
With that our whirlwind tour comes to a close. We've seen how universal algebra gives us tools for exploring the *structure* of things, while universal coalgebra allows us to explore their *behaviour*. Together they gave us a way to rigorously analyse the duality between structure and behaviour. Earlier in the article we made the rather bold claim that this duality transcends the examples we've seen here and goes up all the way to the foundations of thought. We'll end on a similarly dramatic note by giving you a philosophical question to ponder:

> *Is a "thing" best defined by its constituent parts (structure) or by its observable actions(behaviour).*

Algebras (using constructors)

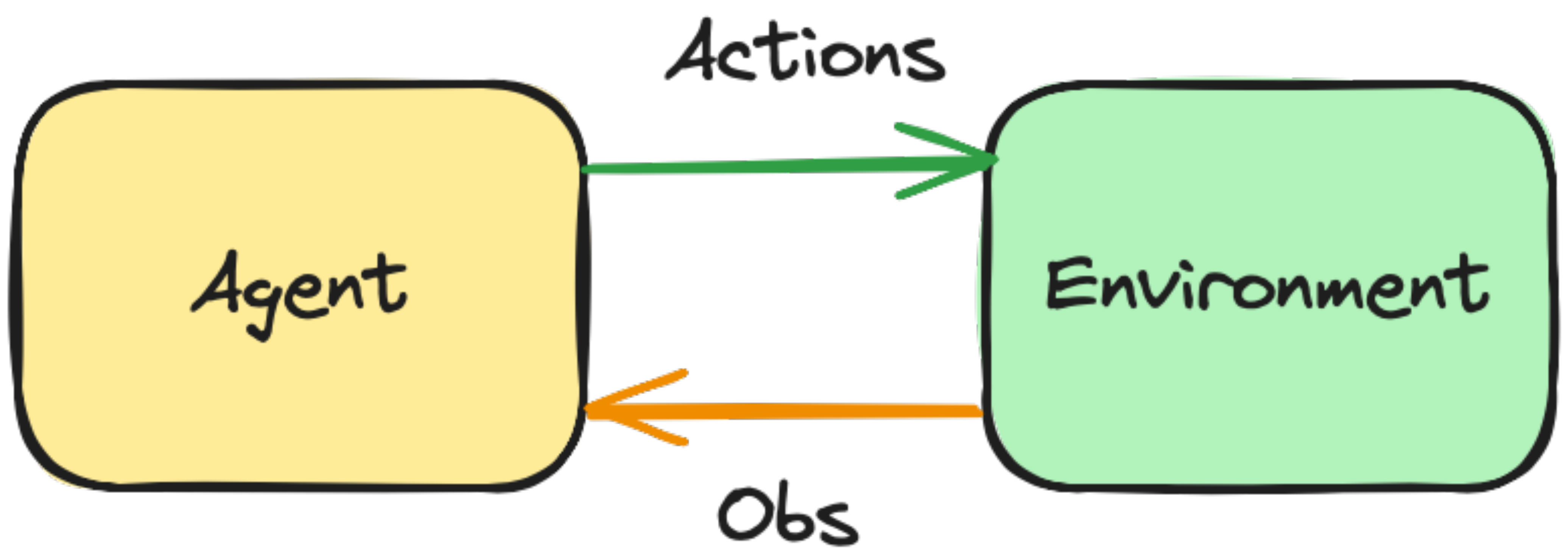
$$(3+1)*(4-2)$$
$$=(4)*(2)$$
$$=8$$

Coalgebras (using destructors)



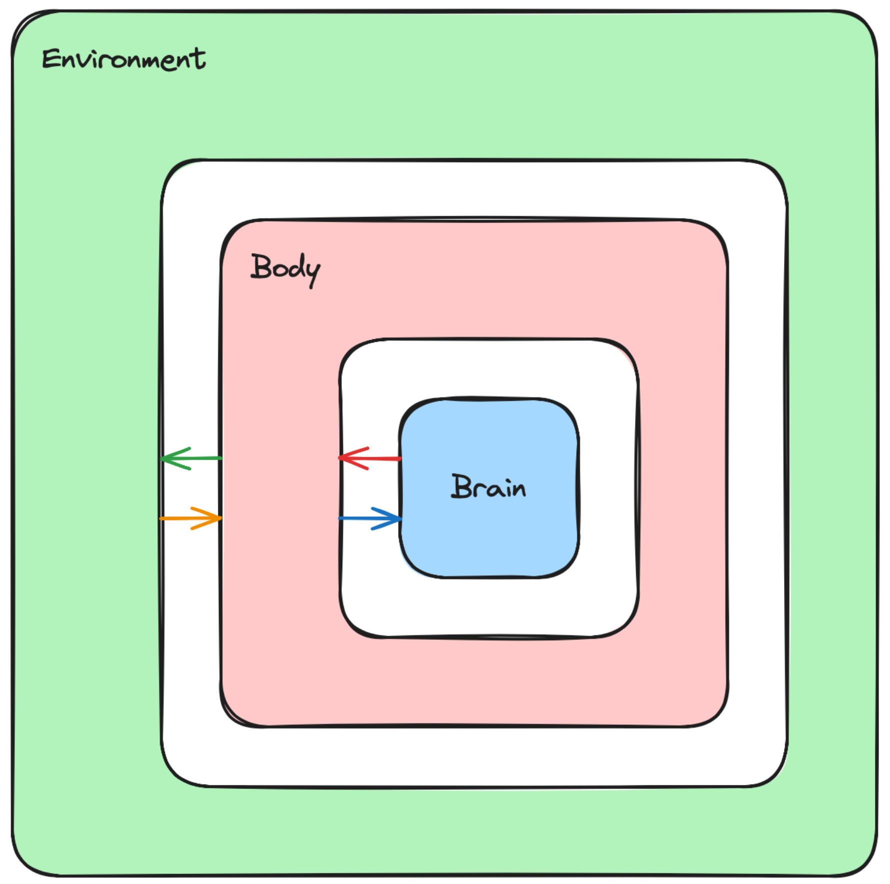
What I am interested in

Background

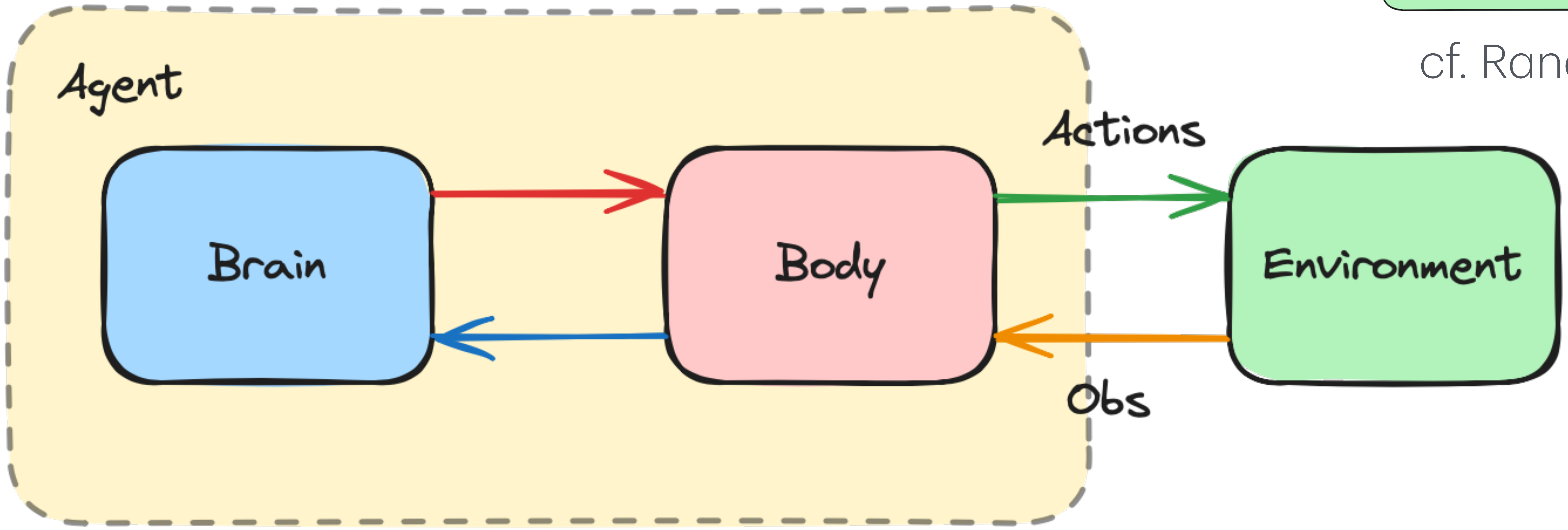


Unpacking that a little

Factorising the agent

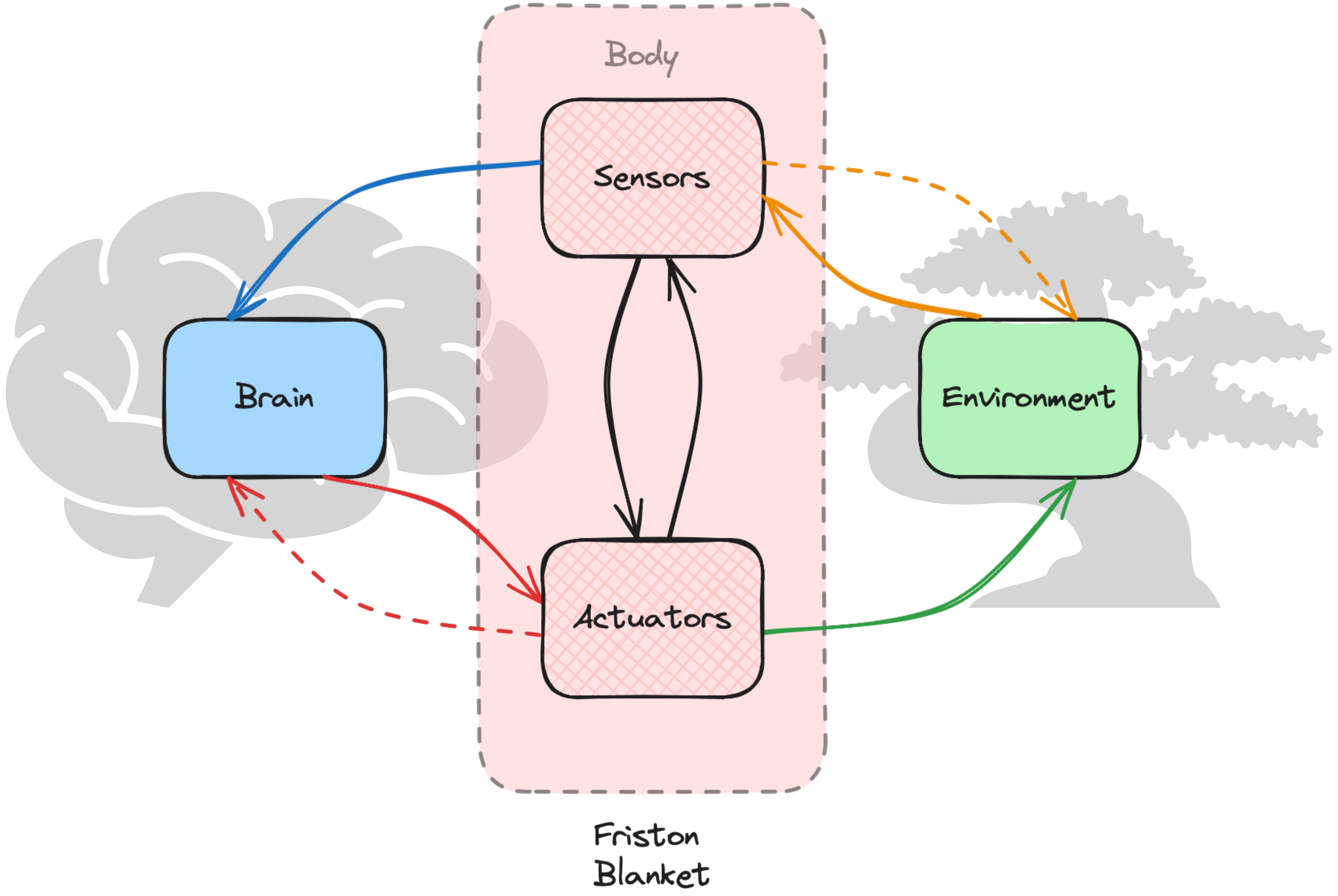


cf. Randy Beer



Meanwhile, the FEP

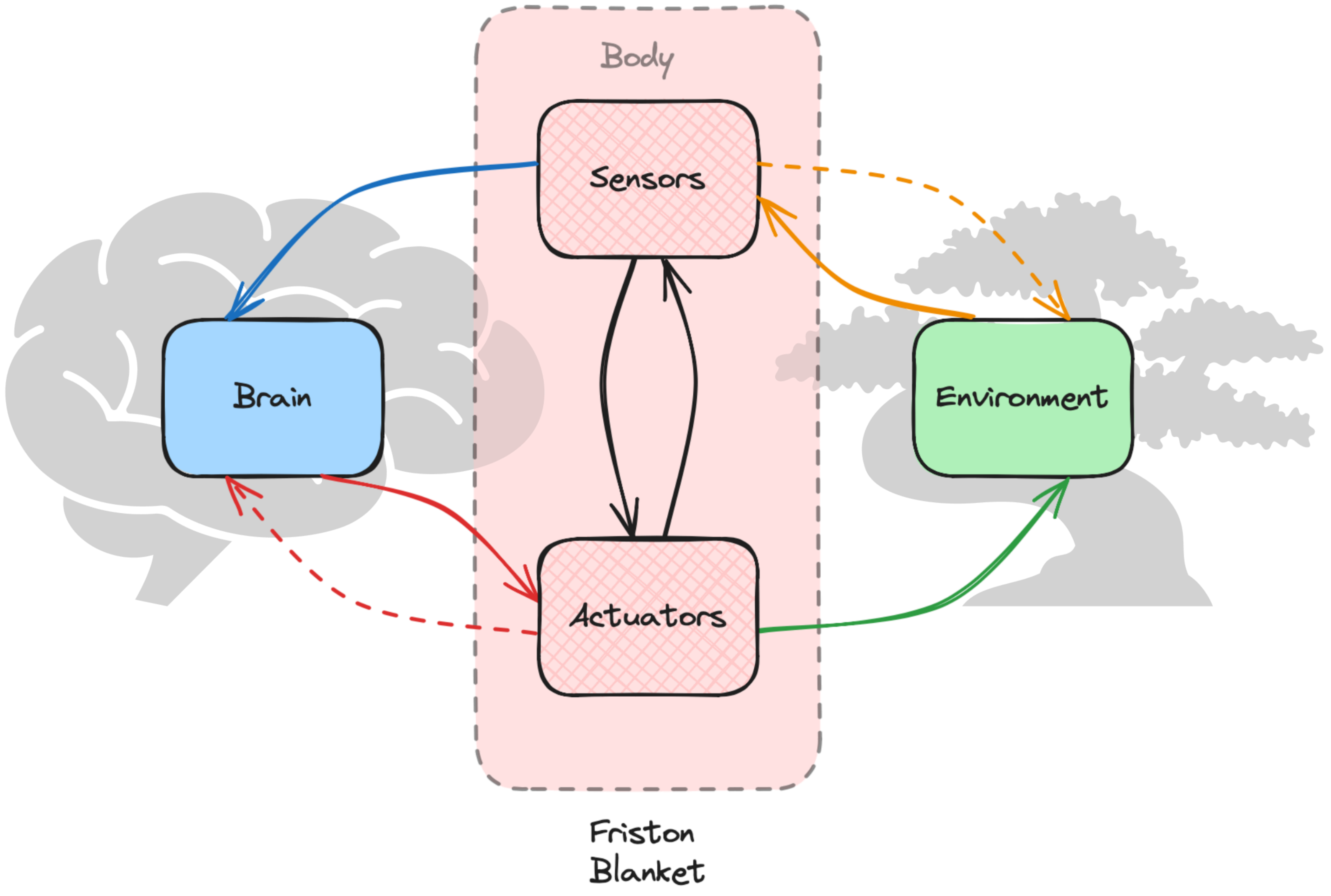
Friston blankets, boundary factored into sensors and actuators



The free energy principle

What agents ARE

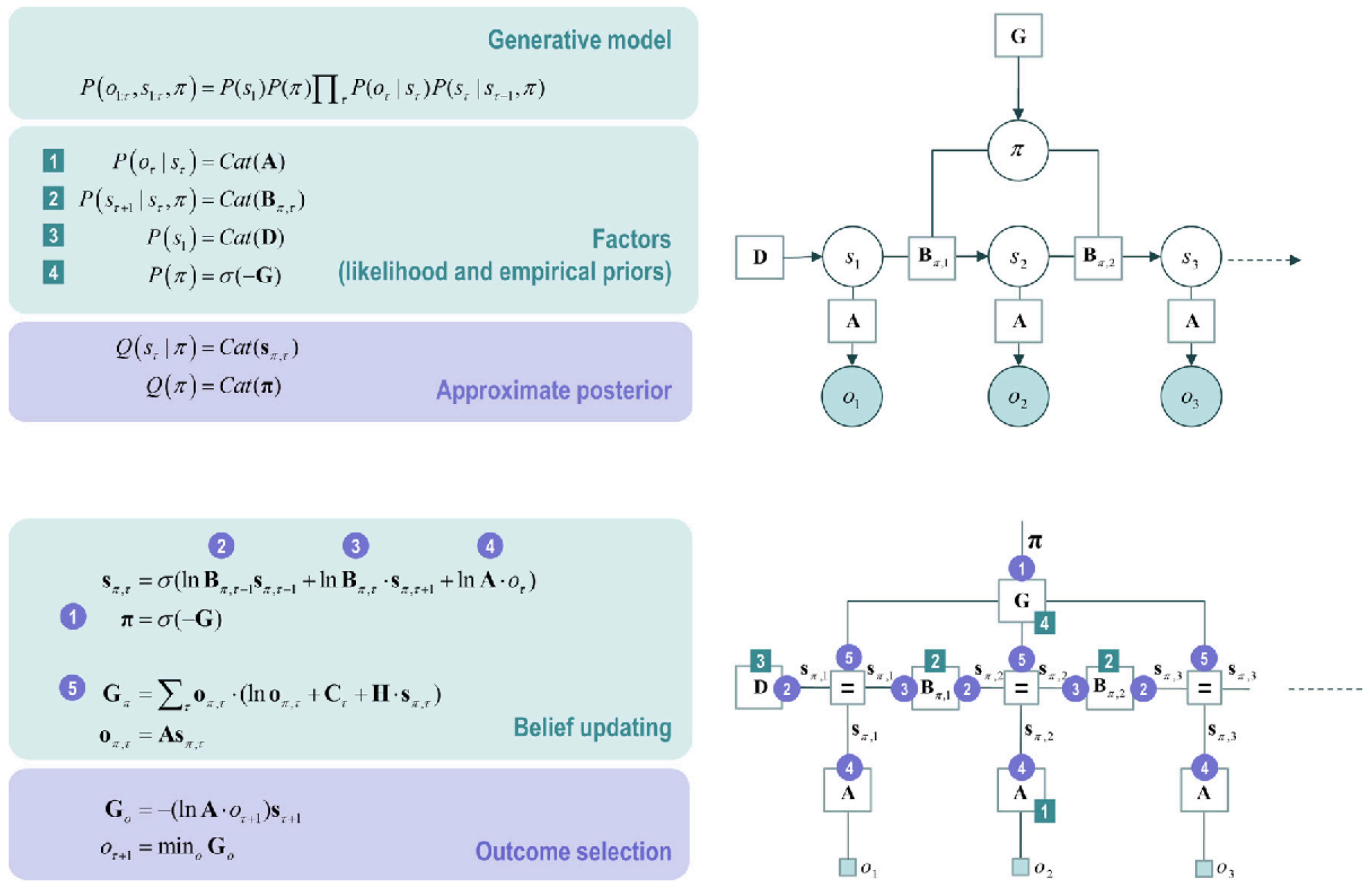
- A foundational theory of agents, (living) systems, “things”
- A thing is a “thing” if and only if it (appears to) minimise(s) free energy
- Friston blankets as a “veil” that separates internal from external states



Active inference

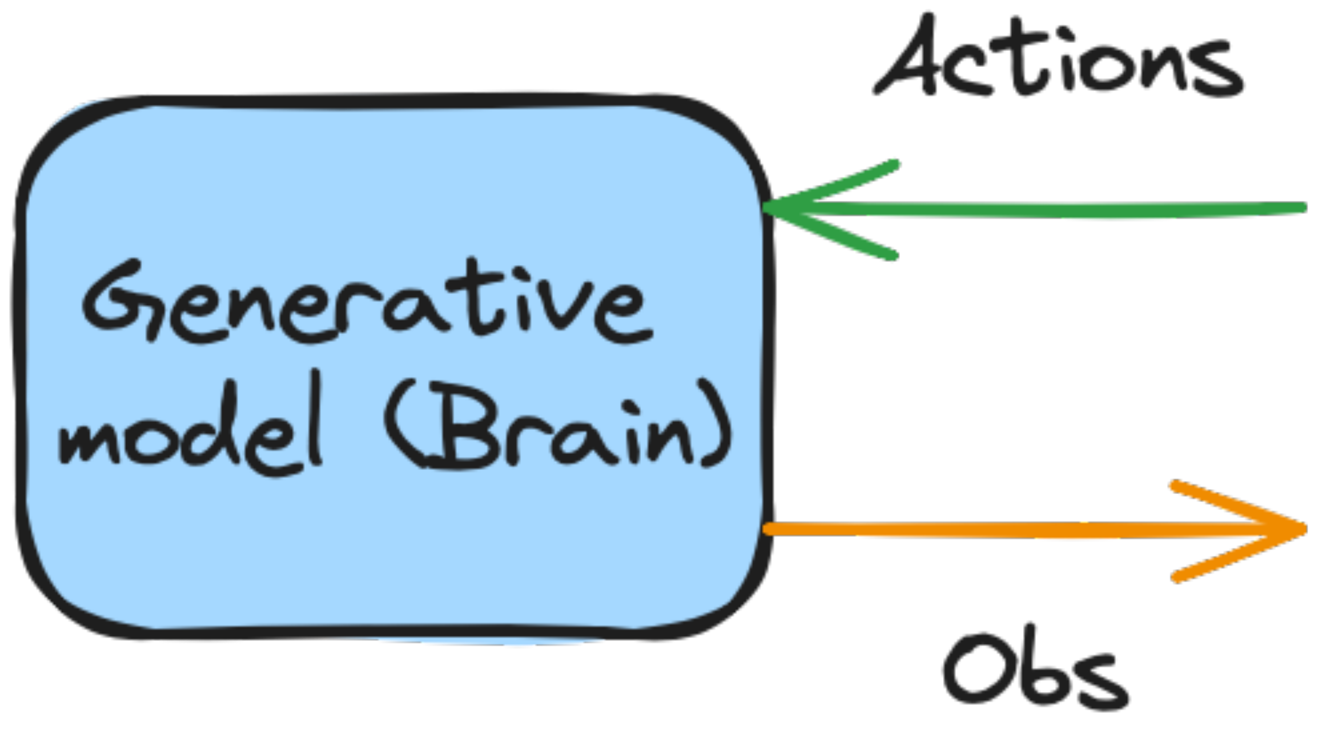
What agents DO

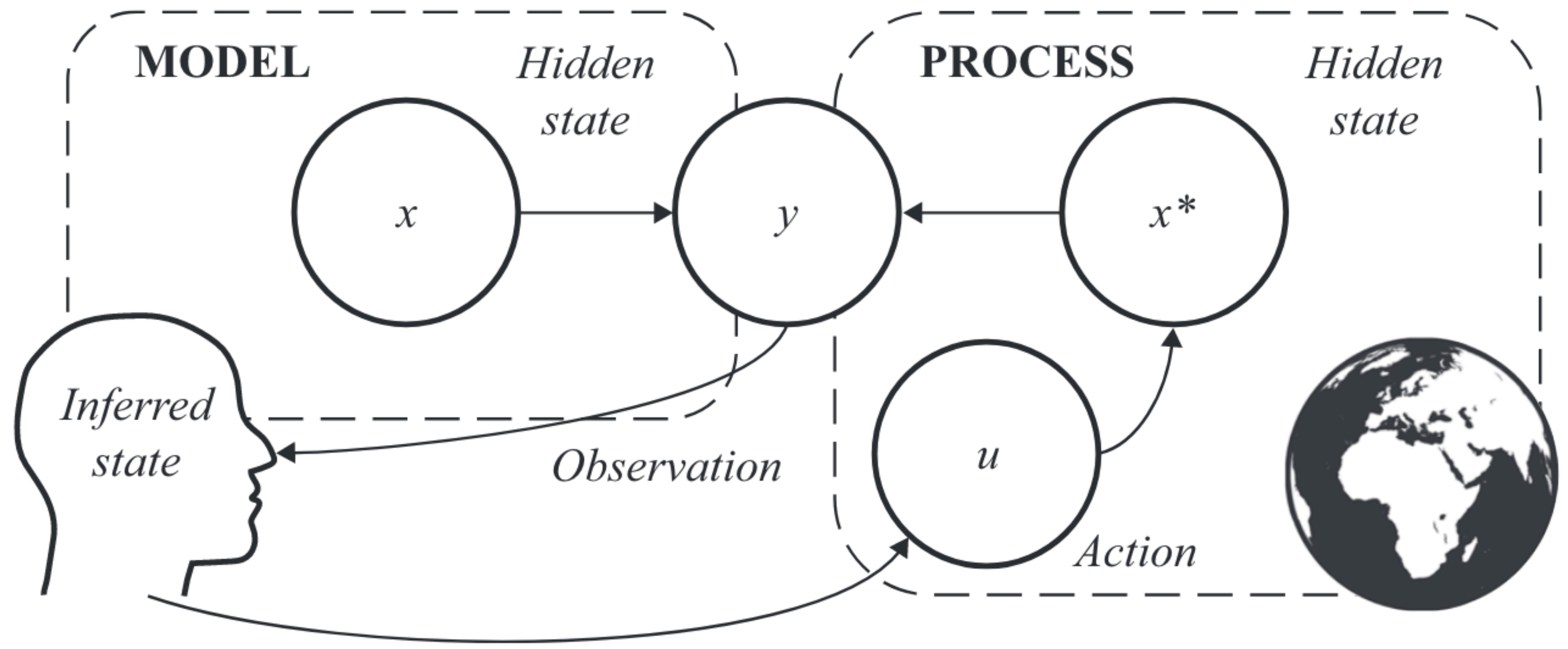
- Assumes POMPDs/state-space models problem structure (~ RL setup)
- Provides an alternative cost function (expected free energy)
- ...ideally one that is derived from the FEP, but it can stand without it

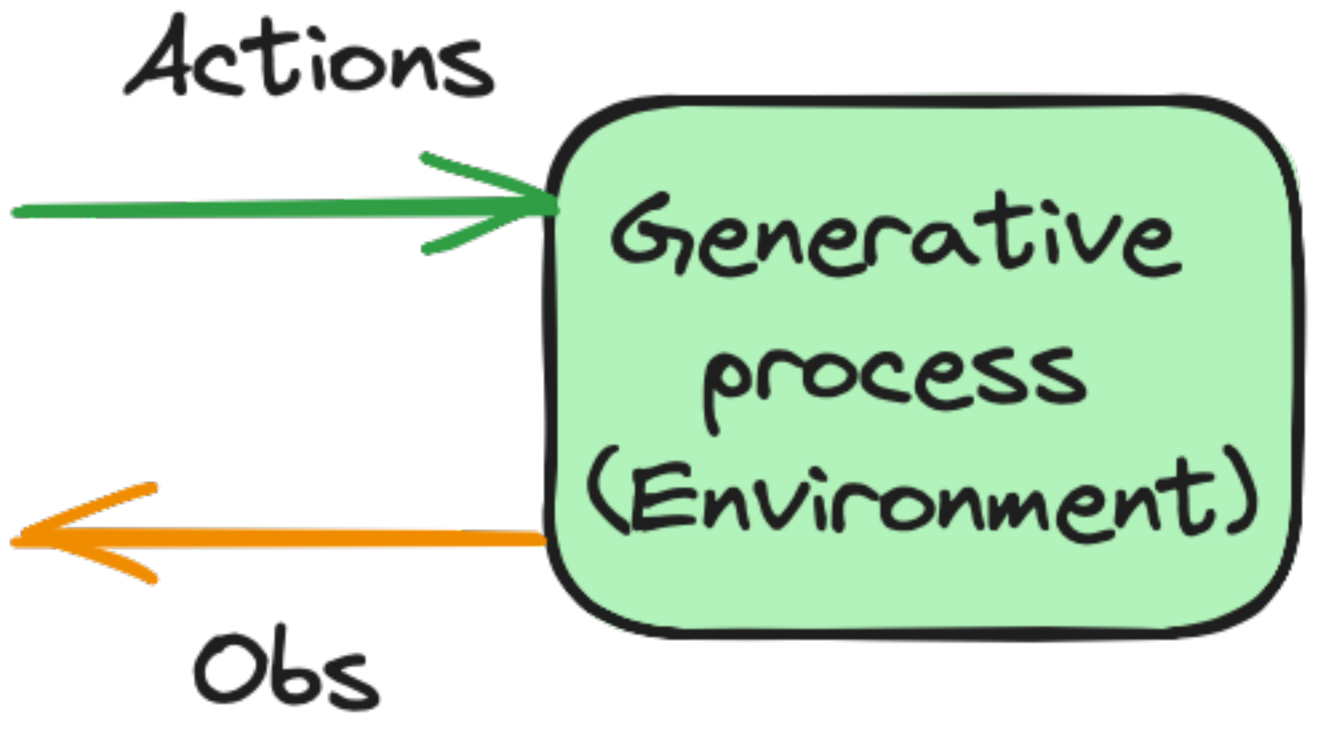


Predictive coding within the FEP

Some intuitive ideas







Parr et al. 2023

Predictive coding “commandments” I will follow:

1. The generative model ought to produce observations consistent with the ones produced by the generative process (create an explanation for the observations)

2. The generative model does not need to be a mirror of the generative process

A 1/2 new perspective

Let's simplify things

- The free energy principle presentation is complicated for, mostly, no reason
- Core ideas behind the FEP have admittedly been proposed in the past (see "internal model principle" next)
- Disentangle problem formulation from algorithms and approximations (let's instead look at the general problem, but most of these ideas can easily be implemented)

4.5.1 A Generative Model for Predictive Coding

To motivate the form of generative model used for continuous states, we start with the following pair of equations:

$$\begin{aligned} \dot{x} &= f(x,v) + \omega_x \\ y &= g(x,v) + \omega_y \end{aligned} \tag{4.15}$$

The first of these expresses the evolution of a hidden state over time, according to a deterministic function ($f(x,v)$) and stochastic fluctuations (ω). The second equation expresses the way in which data are generated from the hidden state. In each case, the fluctuations are assumed normally distributed, giving the following probability densities for the dynamics and likelihood:

$$\begin{aligned} p(\dot{x}|x,v) &= \mathcal{N}(f(x,v), \Pi_x) \\ p(y|x,v) &= \mathcal{N}(g(x,v), \Pi_y) \end{aligned} \tag{4.16}$$

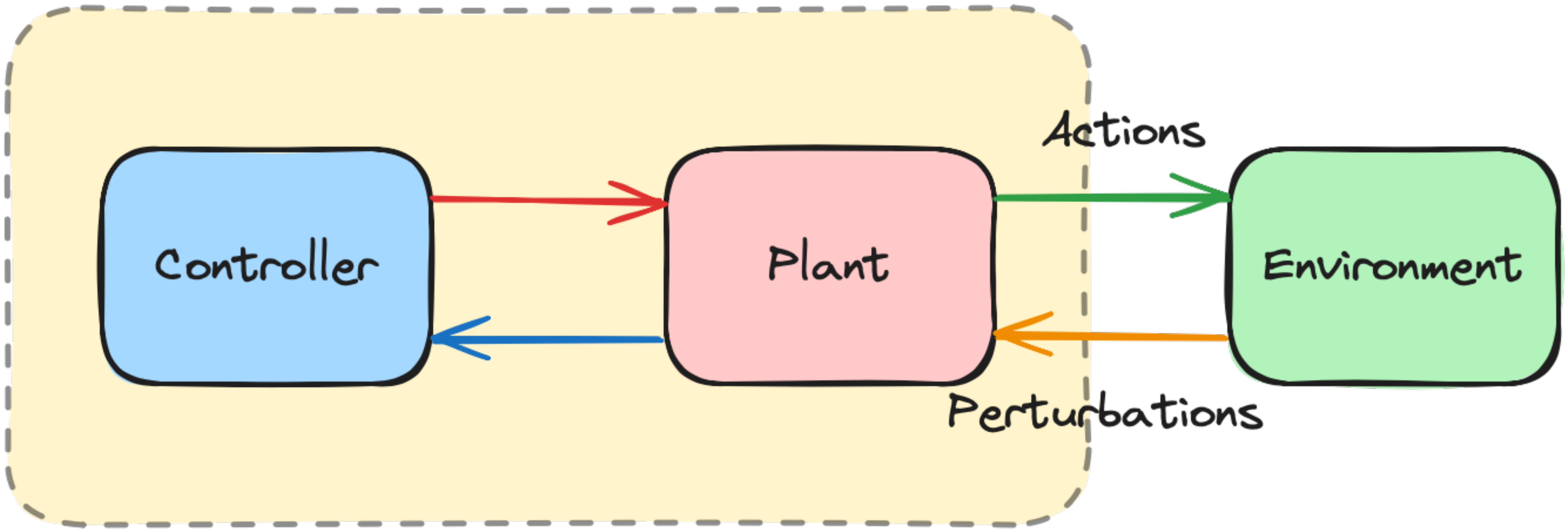
$$\left.\begin{aligned} D\tilde{x} &= \tilde{f}(\tilde{x},\tilde{v}) + \tilde{\omega}_x \\ \tilde{y} &= \tilde{g}(\tilde{x},\tilde{v}) + \tilde{\omega}_y \end{aligned}\right\} \Rightarrow \begin{aligned} p(\tilde{x}|\tilde{v}) &= \mathcal{N}(D\cdot\tilde{f}, \tilde{\Pi}_x) \\ p(\tilde{y}|\tilde{x},\tilde{v}) &= \mathcal{N}(\tilde{g}, \tilde{\Pi}_y) \end{aligned} \tag{4.18}$$

$$\begin{aligned} F[\mu,y] &= -\ln p(\tilde{y},\tilde{\mu}_x,\tilde{\mu}_v) \\ &= \tfrac{1}{2}\tilde{\varepsilon}\cdot\tilde{\Pi}\tilde{\varepsilon} \\ &= \tfrac{1}{2}\left(\tilde{\varepsilon}_y\cdot\tilde{\Pi}_y\tilde{\varepsilon}_y + \tilde{\varepsilon}_x\cdot\tilde{\Pi}_x\tilde{\varepsilon}_x + \tilde{\varepsilon}_v\cdot\tilde{\Pi}_v\tilde{\varepsilon}_v\right) \\ \tilde{\varepsilon} &= \begin{bmatrix} \tilde{\varepsilon}_y \\ \tilde{\varepsilon}_x \\ \tilde{\varepsilon}_v \end{bmatrix} = \begin{bmatrix} \tilde{y} - \tilde{g}(\tilde{\mu}_x,\tilde{\mu}_v) \\ D\tilde{\mu}_x - \tilde{f}(\tilde{\mu}_x,\tilde{\mu}_v) \\ \tilde{\mu}_v - \tilde{\eta} \end{bmatrix} \\ \tilde{\Pi} &= \begin{bmatrix} \tilde{\Pi}_y & & \\ & \tilde{\Pi}_x & \\ & & \tilde{\Pi}_v \end{bmatrix} \end{aligned} \tag{4.19}$$

Parr et al. 2023

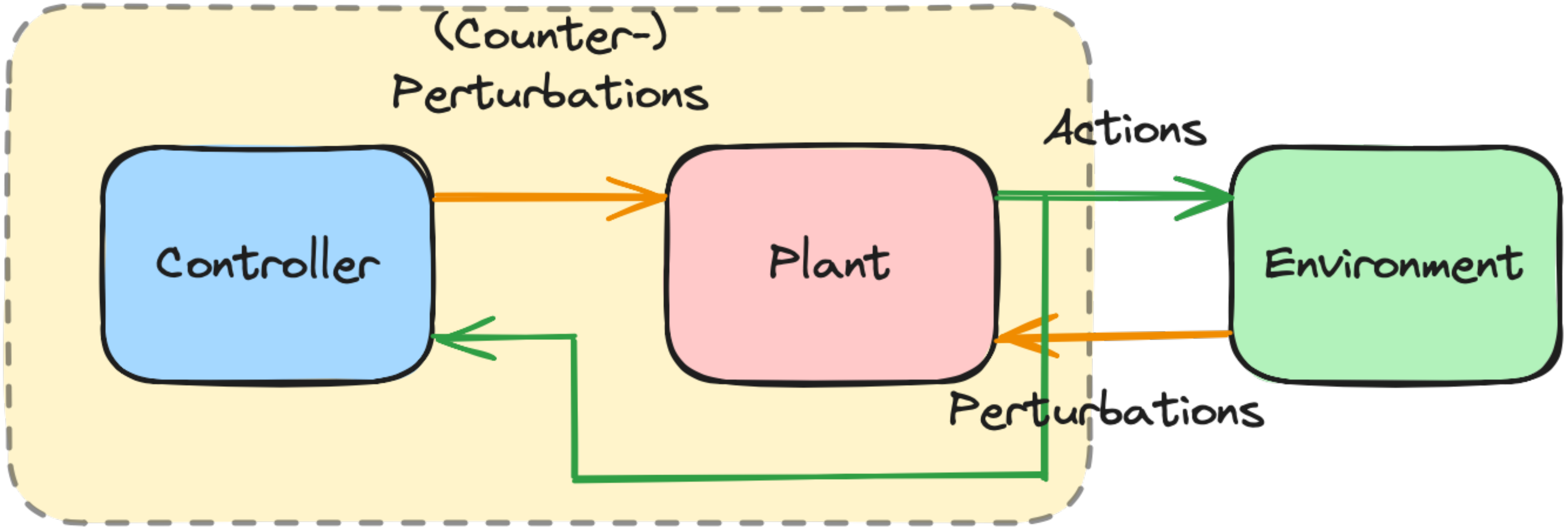
Meanwhile, in control theory

Control-plant-environment factorisation



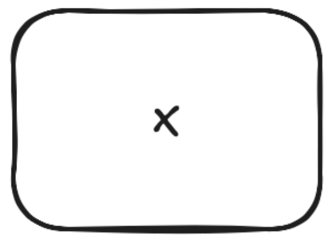
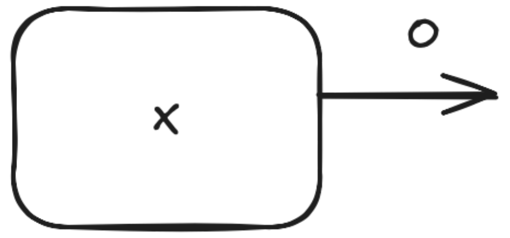
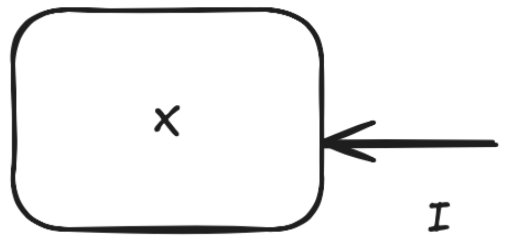
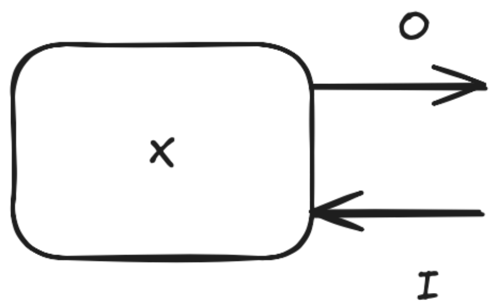
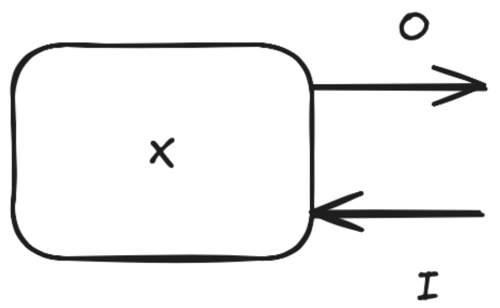
Internal model principle

A model of homeostasis (implying a model?)



Abstracting things

Coalgebras as a language for dynamical systems

	The "standard" way	The coalgebraic way	Graphically (informal)
A (closed) dynamical system	$(X, \alpha : X \to X)$	$(X, f : X \to X)$	
A dynamical system with outputs	$(X, O, \alpha : X \to X,$ $\gamma : X \to O)$	$(X, f_{\mathsf{Out}} : X \to X \times O)$	
A dynamical system with inputs	$(X, I, \beta : X \times I \to X)$	$(X, f_{\mathsf{In}} : X \to X^I)$	
A dynamical system with inputs&outputs	$(X, I, O, \beta: X \times I \to X,$ $\gamma: X \to O)$	$(X, f_{\mathsf{Moore}} : X \to X^I \times O)$	
A probabilistic system with inputs&outputs	$(X, I, O, \beta_P: X \times I \to P(X),$ $\gamma_P: X \to P(O))$	$(X, f_{\mathsf{PrMoore}} :$ $X \to P(X)^I \times P(O))$	

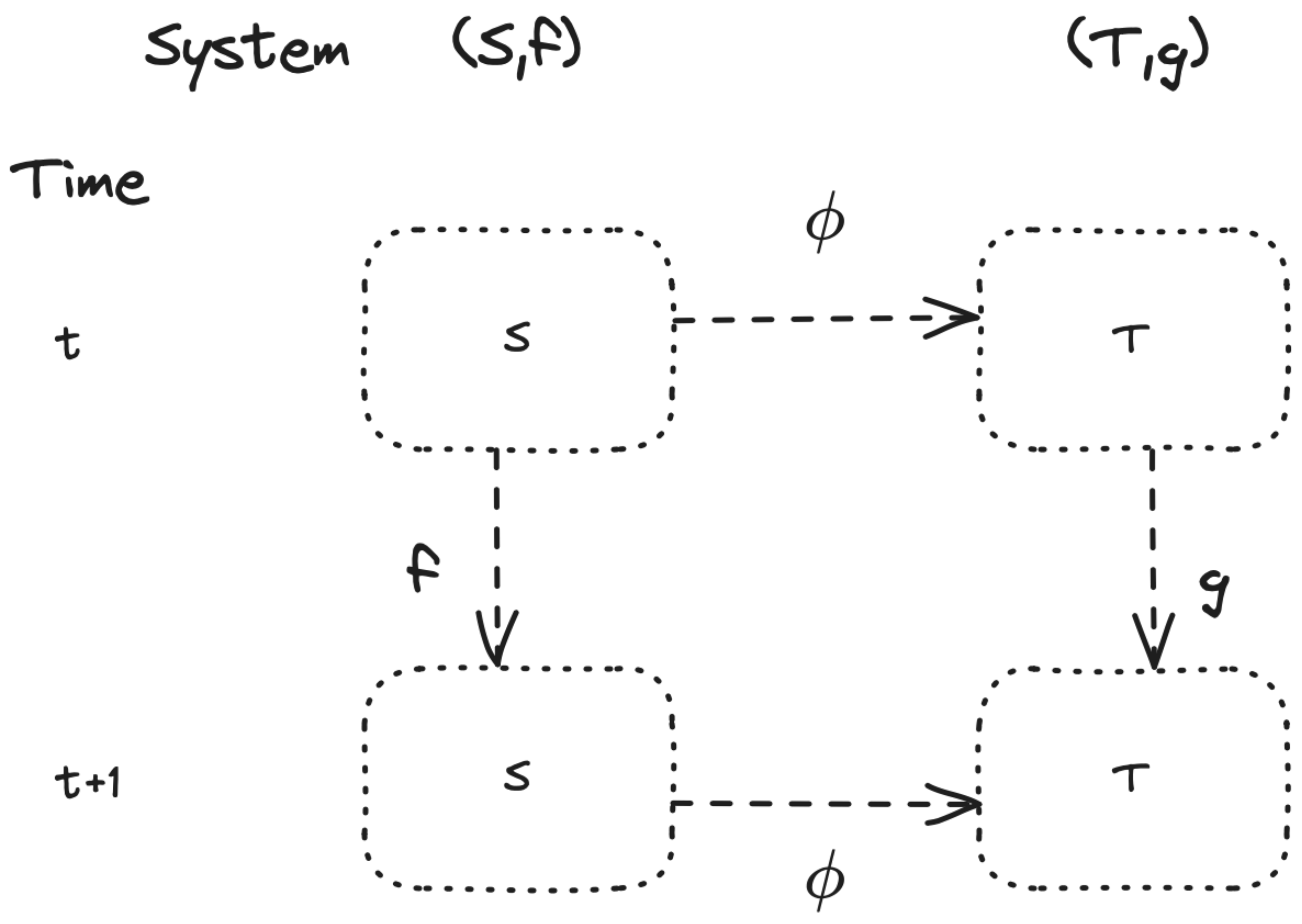
Maps between (closed) systems

Coalgebra (homo)morphisms by example

Take two closed systems, (S, f) and (T, g). A map between these systems is a function ϕ such that the following diagram commutes

$$\begin{tikzcd} S \arrow[r, "\phi"] \arrow[d, "f"'] & T \arrow[d, "g"] \\ S \arrow[r, "\phi"'] & T \end{tikzcd}$$

or in other words, if $g(\phi(S)) = \phi(f(S))$.



Maps between (open) systems

Coalgebra (homo)morphisms by example

Take two probabilistic dynamical systems, (S, f_{PrMoore}) and (T, g_{PrMoore}). A map between these systems is a function ϕ such that the following diagram commutes

$$\begin{tikzcd}
S \arrow[r, "\phi"] \arrow[d, "f_{\text{PrMoore}}"] & T \arrow[d, "g_{\text{PrMoore}}"] \\
P(O) \times P(S)^I \arrow[r, "P(\mathrm{id}_O) \times P(\phi)^I"'] & P(O) \times P(T)^I
\end{tikzcd}$$

or in other words, if $g_{\text{PrMoore}}(\phi(S)) = \phi(f_{\text{PrMoore}}(S))$.

(Same thing as before, but requiring that (T, g_{PrMoore})'s inputs and outputs are equal to (S, f_{PrMoore})'s at each time step whenever there's a map between their states that commutes with the systems' dynamics.)

Behavioural equivalence

Bisimulations, congruences on *behaviour*, by example

Take two closed systems, (S, f) and (T, g). A bisimulation between these systems is a relation R such that the following diagram commutes

$$
\begin{tikzcd}
& S \times T & \\
& R \arrow[u] \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] \arrow[dd, dashed, "\exists\ \gamma"] & \\
S \arrow[dd, "f"'] & & T \arrow[dd, "g"] \\
& R \arrow[dl, "\pi_1"] \arrow[dr, "\pi_2"'] & \\
S & & T
\end{tikzcd}
$$

or in other words, if $g(\pi_2(R)) = \pi_2(\gamma(R))$ and $f(\pi_1(R)) = \pi_1(\gamma(R))$.

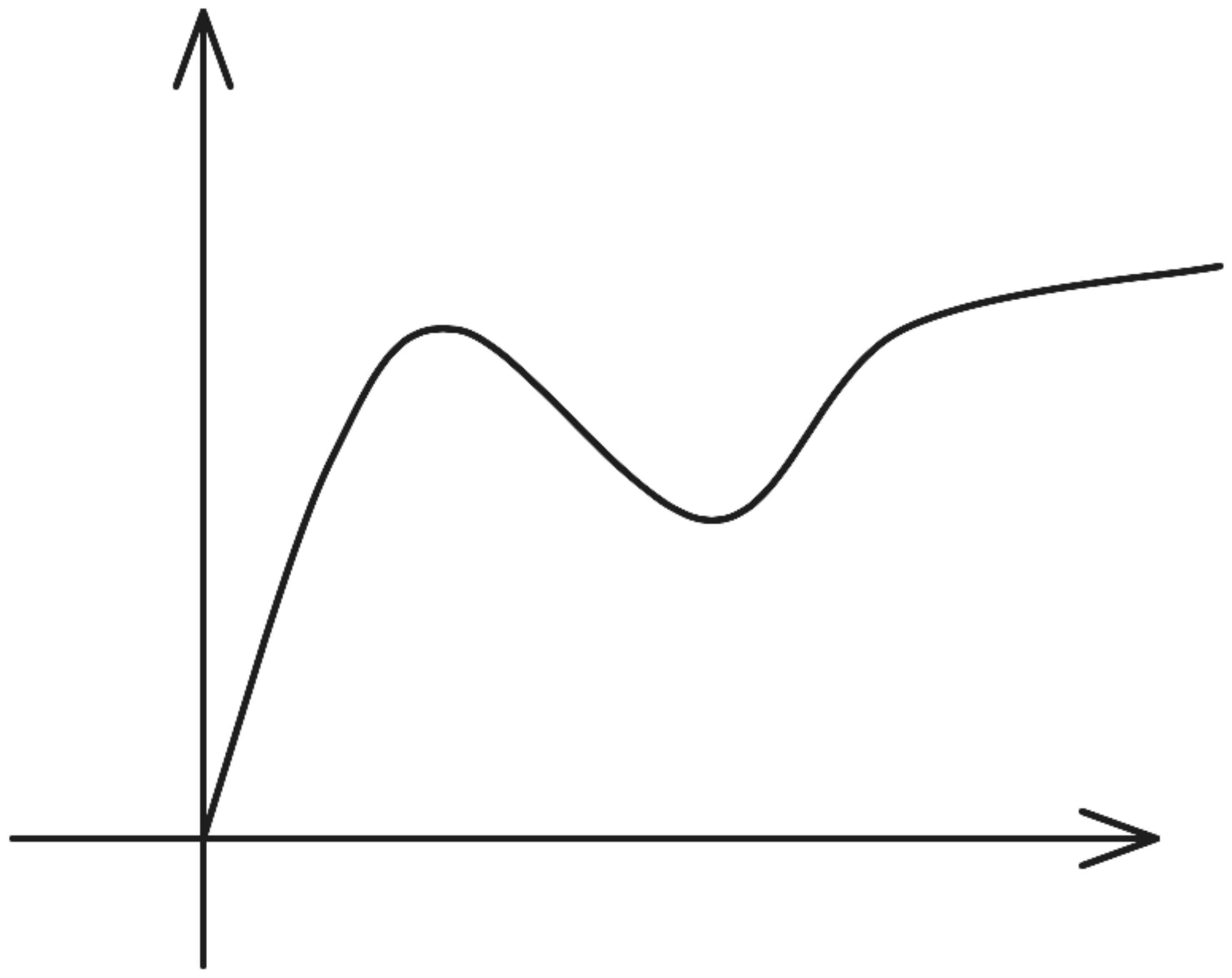
Take two open systems, $(S, f_{\mathsf{PrMoore}})$ and $(T, g_{\mathsf{PrMoore}})$. A bisimulation between these systems is a relation R such that the following diagram commutes

$$
\begin{tikzcd}
& S \times T & \\
& R \arrow[u] \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] \arrow[d, dashed, "\gamma_{\mathrm{PrMoore}}"] & \\
S \arrow[d, "f_{\mathrm{PrMoore}}"'] & P(R)^{In} \times P(Out) \arrow[dl, "P(\pi_1)^{In} \times P(\mathrm{id}_{Out})"] \arrow[dr, "P(\pi_2)^{In} \times P(\mathrm{id}_{Out})"'] & T \arrow[d, "g_{\mathrm{PrMoore}}"] \\
P(S)^{In} \times P(Out) & & P(T)^{In} \times P(Out)
\end{tikzcd}
$$

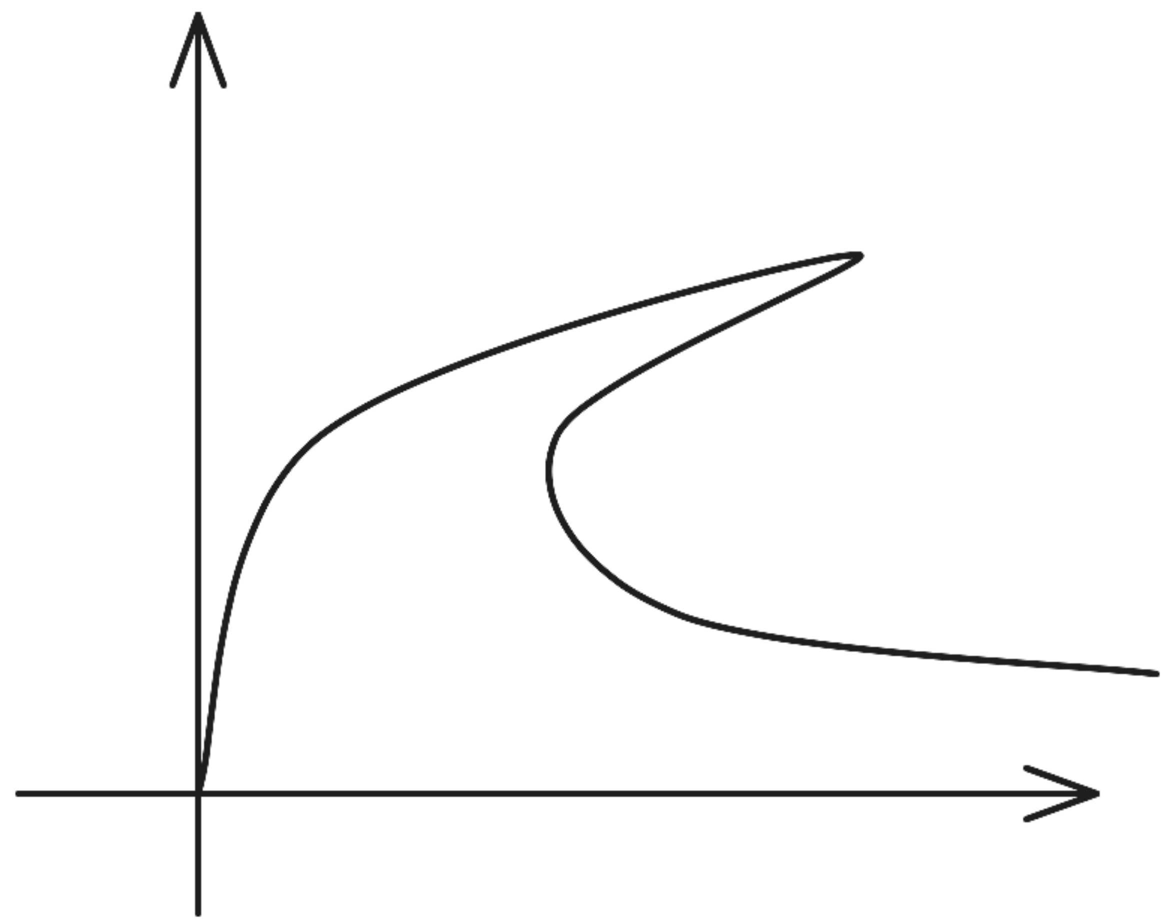
Maps of systems vs. bisimulations?

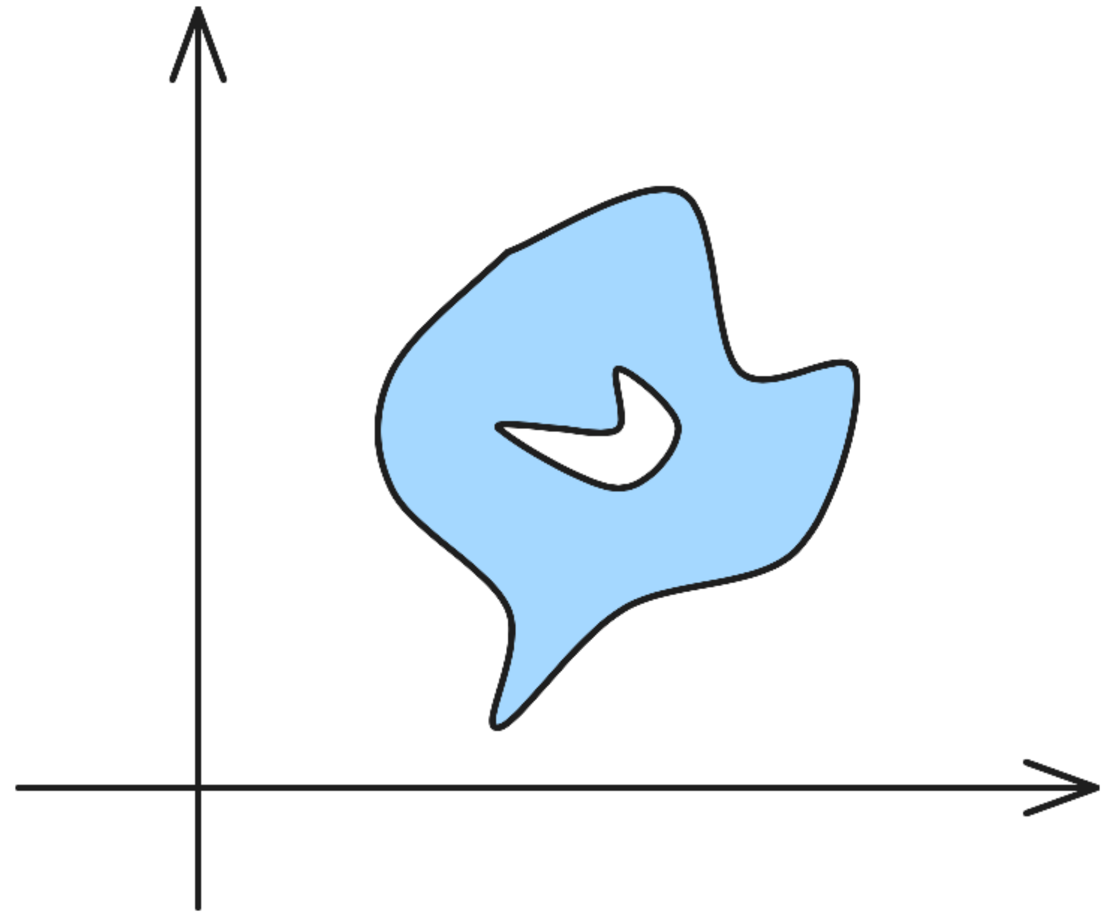
Functions vs relations in Set

Maps of systems vs. bisimulations in Coalg(Set)



vs.





What is predictive coding *mathematically*?

Proposal: predictive coding as bisimulations

Predictive coding "commandments" I will follow:

1. The generative model ought to produce observations consistent with the ones produced by the generative process (create an explanation for the observations)

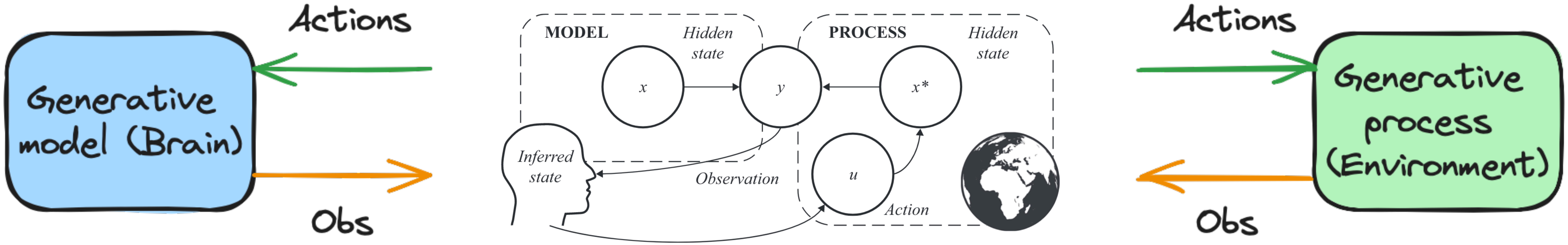
—> There is a bisimulation between a "correct" generative model and a given generative process

2. The generative model does not need to be a mirror of the generative process

—> Agents only need an understanding of generative processes as relevant for their tasks/actions repertoire

1. The relation between agent and environment

Agent-environment attunement as a bisimulation



$$\begin{tikzcd}
 & & S_{\text{Brain}} \times S_{\text{Env}} & & \\
 & & R_{\text{Homeostasis}} \arrow[u] \arrow[dll] \arrow[drr] \arrow[dd] & & \\
S_{\text{Brain}} \arrow[dd] & & & & S_{\text{Env}} \arrow[dd] \\
 & & P(R_{\text{Homeostasis}})^{Act} \times P(Obs) \arrow[dll] \arrow[drr] & & \\
P(S_{\text{Brain}})^{Act} \times P(Obs) & & & & P(S_{\text{Env}})^{Act} \times P(Obs)
\end{tikzcd}$$

2. "Action-oriented" models

A way to look at compressed models

"World models" meaning "models the environment" is a pretty flashy but bad name

Surely they can't be about the entire universe dynamics, so what are they talking about?

Action oriented models seem more reasonable (but not formal):

> between brain, body, and world. Neural representations, this work has suggested, are not action-neutral mirrors of the world. Instead they are in some deep sense 'action-oriented' (Clark 1997, Engel et al. 2013). They are geared to promoting successful, fast, fluent actions and engagements for a creature with specific needs and bodily form. Such representations will be as minimal as possible, neither encoding nor processing information in costly ways when simpler routines, combined with world-exploiting actions, can do the job.

Clark 2015

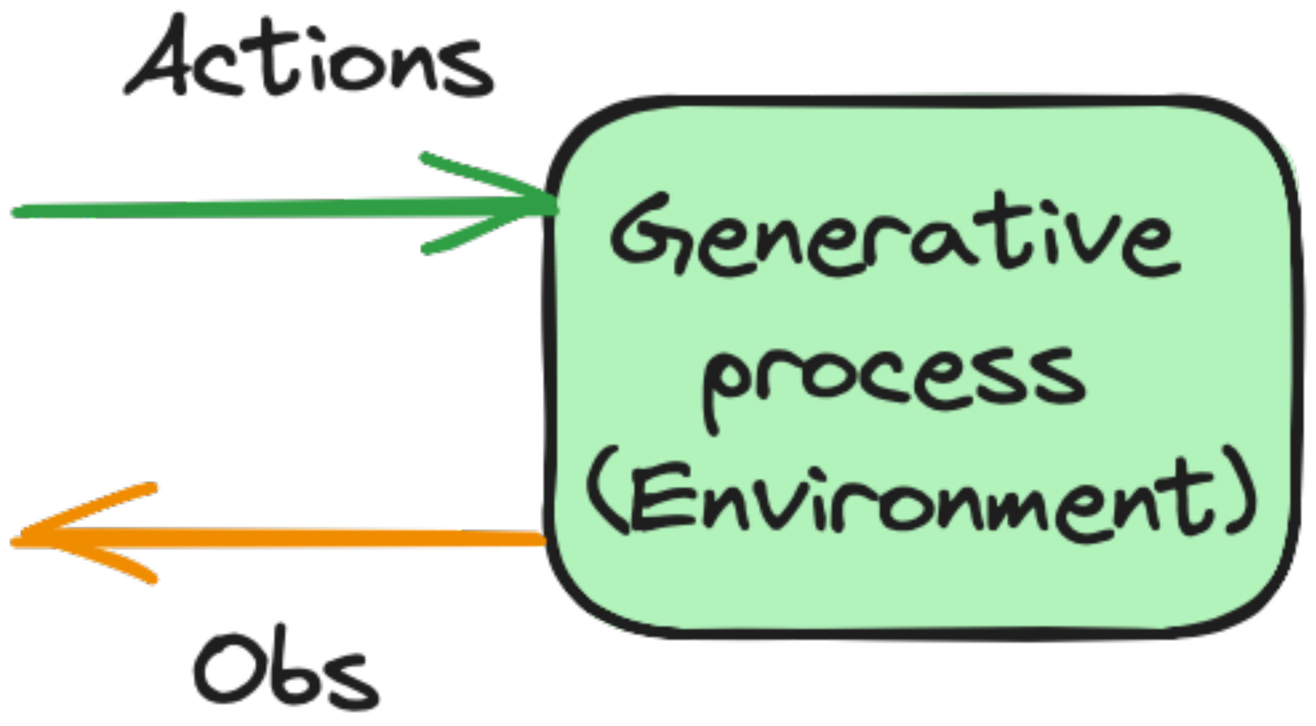
Proposed formalisation: bisimulation equivalences.

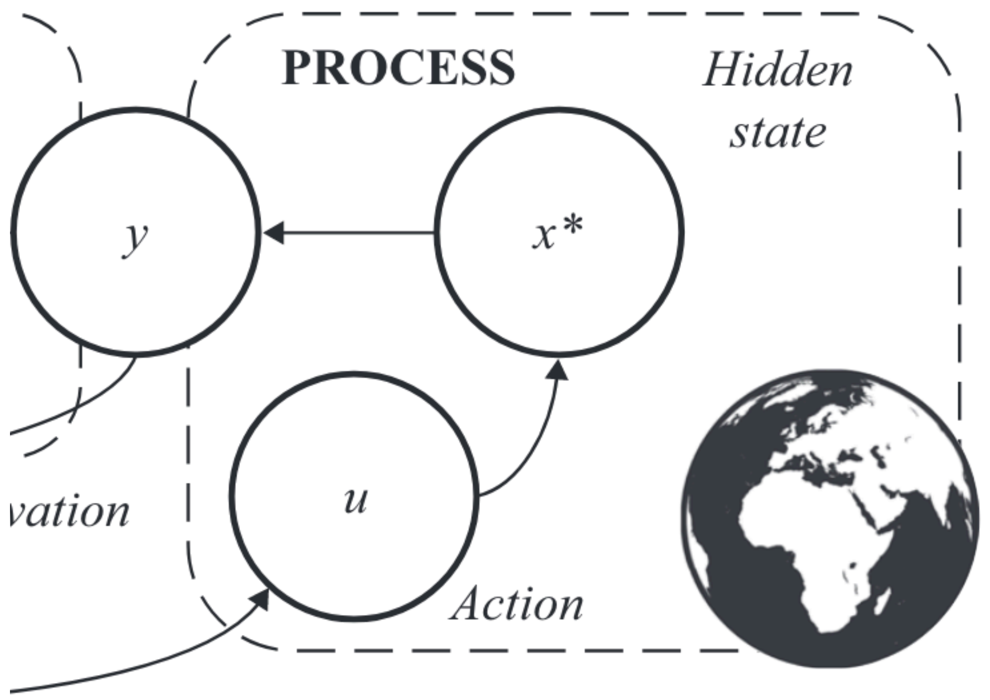
These build (dynamical) compressions of environments, with various possible criteria, for instance:

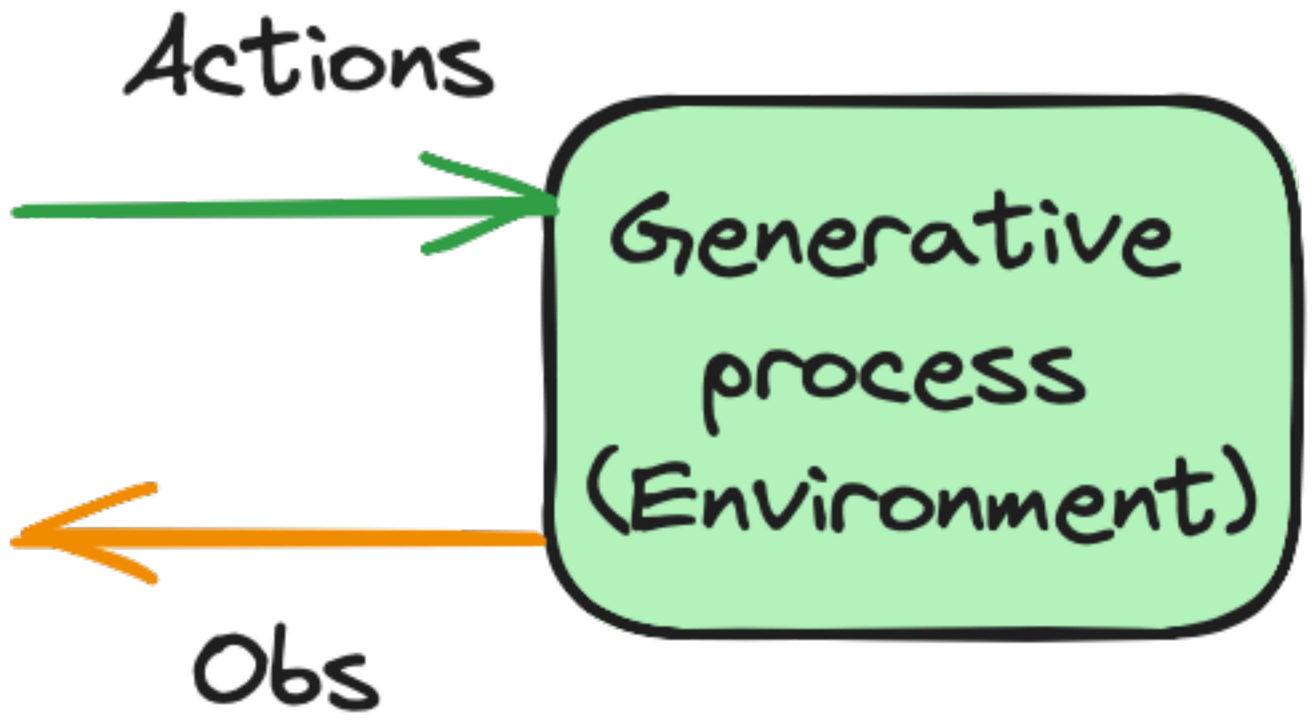
- compression for all possible actions of all possible agents
- compression for all possible actions of a single agent
- compression for all possible actions of a single agent, given the same reward
- compression for the actions of a policy chosen by an agent, given the same reward
- ...

Compressing environments' models

Bisimulation equivalences of environments for a particular goal







$$\begin{tikzcd}
 & S_{\text{Env}} \times S_{\text{Env}} & \\
 & R_{\text{Compress}} \arrow[u] \arrow[dl] \arrow[dr] \arrow[dd] & \\
S_{\text{Env}} \arrow[dd] & & S_{\text{Env}} \arrow[dd] \\
 & P(R_{\text{Compress}})^{Act} \times P(Obs) \arrow[dl] \arrow[dr] & \\
P(S_{\text{Env}})^{Act} \times P(Obs) & & P(S_{\text{Env}})^{Act} \times P(Obs)
\end{tikzcd}$$

Where is embodiment in this story?

The role of the body in standard predictive coding stories

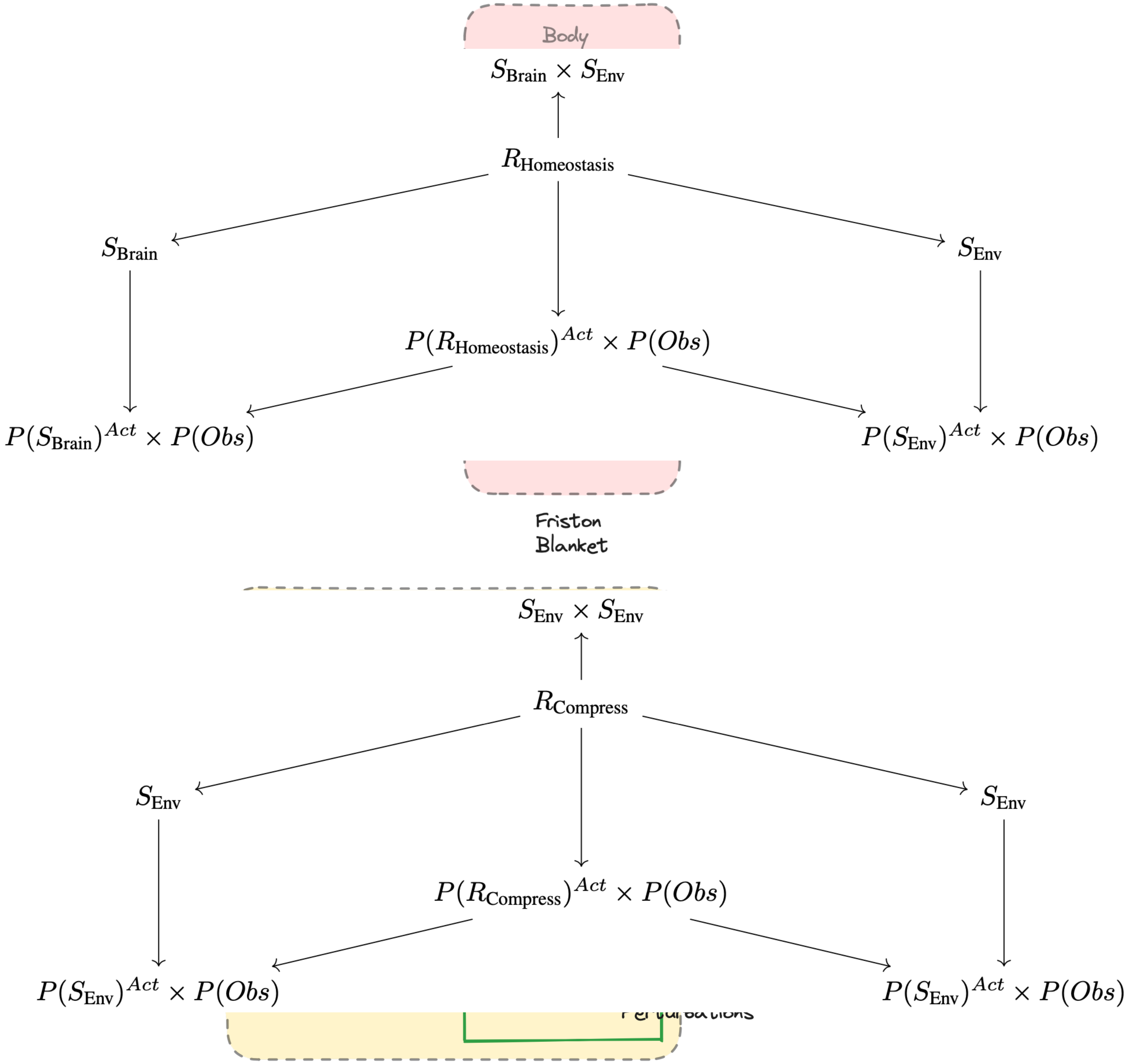
- This is still vastly brain centric (the brain predicts, the brain matches the environment, etc.) but

- ... is it fair to compare “plant” and “body”? Maybe a “controller” is better suited to represent “brain+body” and “plant” should be seen only as “membrane”?

- Does “rewiring action and observations” in the plant/body count as embodiment?

- Does “mirroring the environment” (behaviourally rather than structurally) in the brain count towards embodiment?

- Does “compressing the environment” count towards embodiment?



Summary

- Agent-environment setup
- Coalgebras, one way to talk about behaviour (functionalist-like?)
- Maps between coalgebras to define behavioural equivalence (bisimulations)
- Claims:
  - FEP-style agents captured by bisimulation between brain and environment
  - Better FEP-style agents “compress” the environment to their needs
- The role of the “body” in FEP-style agents seems quite limited