A Characterization of Processing Loops in AI and Biological Systems and its Implications for Understanding Consciousness

Malcolm J. Lett1

1No affiliation

**\* Correspondence:**Malcolm Lett  
malcolm.lett@gmail.com

Keywords: Consciousness, Meta-cognition, Meta-management, Biosemiotic Process, Recurrent Process

Abstract

todo

# Introduction

Why do we have subjective experience of our perceptions and of our mind's thoughts? This question can be broken into three distinct sub-questions. Firstly, what are the mechanisms underlying the cognitive processing associated with subjective experience? Secondly, why do those processes produce the effect of subjective experience, whereas other processes do not? And thirdly, what functional purpose does subjective experience carry over and above similar processes that are not associated with subjective experience?

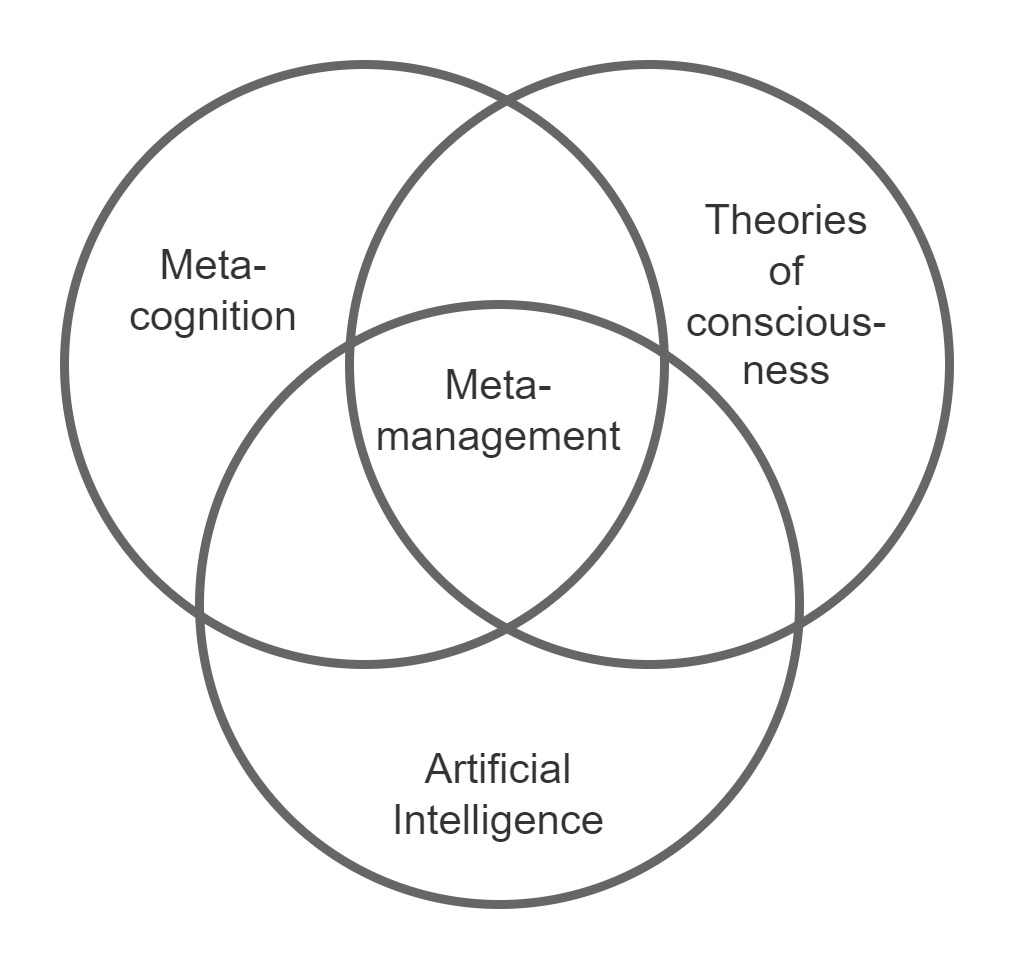
A number of theories of consciousness (TOCs) have been developed, which would attempt to answer some of those questions. The most detailed are based on a computational model of the brain. .... These theories focus primarily on the first question of subjective experience - what are the underlying mechanisms? They also attempt to answer the second question of subjective experience.... But they rarely delve into the third question. Instead, usually they make vague references to the need for adaptation in complex environments.

But what is adaptation? And what ...........

Consciousness is often touted as being there to support adaptation. TOCs don't really cover how they support adaptation at low level. Also, it's not clear what adaptation really means at the low level. Meta cognition covers one particularly pertinent area -knowledge of knowledge and processes. It would appear to support adaptation. But it doesn't cover the low level mechanisms.

Complex agents in complex envs need complex meta management systems, because they have loops. ....

todo .... bla bla, Deep AI has been making great strides, but it's stuck, and needs something new to make it more flexible.

Figure 1: Meta-management as key to meta-cognition, theories of consciousness, and AI

This paper makes the claim that by applying a design stance to connectionist AI, we will discover the management systems underlying those complex agents, and show that it necessitates self referential awareness and control, and that this is at the heart of subjective awareness. To explain that, I will build up the case from first principles, and look at exactly what those low level mechanisms may be. This strengthens the existing claims that meta-cognition is fundamentally intertwined with consciousness, and it provides a link between the lower-level mechanisms described within TOCs and the higher-level observations of meta-cognition.

The rest of this paper is organized as follows. A brief summary of research programmes into meta-cognition and computational theories of consciousness is presented with some of the issues faced by them. A case is then presented for the specific needs of meta-management within computational models. This is followed by a discussion of potential architectures, drawing out their relative strengths and weaknesses. Finally, those architectures are related back to the study of meta-cognition and consciousness, and an argument is presented for the architectures underlying human meta-cognition. The paper concludes by discussing future directions and speculating on the development of artificial systems that employ these architectures.

Todo - table of terms.

* Experiential awareness. Rather than using the word "consciousness", to be more specific, this paper prefers the term "experiential awareness".......
* meta-cognition
* meta-management
* micro and macro recurrency.
* first-order processes
* state-machine
* state
* computational process

# Introduction

....to redo....

..

...what am I really trying to say here?

...meta-cognition has issued, theories of consciousness have issues, I'm interested in consciousness. I can help both by linking these high-level theories to lower-level mechanisms. And to wrap it up, I'll circle back to meta-cognition via semiotics, and suggest how that shows the link to consciousness.

...I haven't introduced anything about meta-cognition, and haven't really suggested anything about consciousness in the intro yet, so it's going to be hard to tie back to those in both the end of the intro and the main content.

..Fundamentally I don't want to write a technical paper about meta-management, although that could be the paper's focus on its own. I want to add to consciousness research. But then, I originally created the Visceral Loop as a general purpose model.

I could stop midway then: meta-cognition. Half-way between pure technical and consciousness. End by tieing back to meta-cognition's struggles with identifying what's meta-cognitive and what's first-order. Using consciousness as the measuring device, rather than studying it specifically, we can show that lab results help us identify which architecture is plausible, and thus help to discriminate within meta-cognitive research.

..

# Meta-cognition

In ..{year}.. it was found that people who had a better understanding of their own learning abilities, learned better. People with more awareness of their learning abilities developed better learning strategies to leverage their strengths, while working around their limitations (eg: using mnemonics to improve memory). Thus the field of *meta-cognition* was born - to study the mechanisms whereby people can monitor their own mental behaviors and use that knowledge for adaptation. ......need to list some lab-observed behaviors......

Meta-cognition is defined as knowledge about one's own knowledge.....

The study of meta-cognition has a particular relevancy to the study of consciousness. Why do we have conscious experiential awareness of our external perceptions? Why do we have conscious experiential awareness of some aspects of our own mind's state (eg: inner thoughts)? Theories attempting to explain the evolutionary advantage of this conscious awareness generally implicate adaptive flexibility. But they fail to explain precisely what kinds of adaptive flexibility need conscious awareness, and why conscious awareness is needed for those adaptive flexibilities. The study of meta-cognition provides insight because it specifically addresses questions around our ability to know aspects of our own mind's state.

But meta-cognition studies are embroiled in debate about which lab-observed behaviors are truly meta-cognitive. Many of the claimed behaviors might be explained by unconscious processes. And lab results are hard to interpret – eg: the difficulty is separating activated brain regions from involvement in the original meta-cognition versus the production of verbal report? Thus, a deeper understanding of the low-level mechanisms underlying meta-cognition would help significantly to untangle the confusion.

*todo -* Define: first-order processes.

Meta-cognition has been variously studied in terms of so called "feelings of knowing" where one thinks they know the answer before recalling the answer itself (Rosenthal, 2012; Shimamura, 2000; Metcalfe & Shimamura, 1994), memory of the source of knowledge or other memories (Dunlosky & Bjork, 2008; Shimamura, 2000; Fernandez-Duque, 2000; Bejamin et al, 1998; Metcalfe & Shimamura, 1994), judgements of certainty and error detection (Carruthers & Williams, 2022; Cleeremens, 2020; Whitmarsh, Oostenveld, Almeida & Lundqvist, 2017; Fernandez Cruz et al, 2016; Paul et al, 2015; Fleming et al "Metacognition..." 2012; Fleming et al "Prefrontal..." 2012; Shimamura, 2000; Fernandez-Duque, 2000), classification of first-order outcomes into knowledge, hope, fear, regret, etc. (Cleeremans et al, 2007), identification of links between separately obtained knowledge (Clark & Karmiloff-Smith, 1993; Karmiloff-Smith, 1992), representing the absence of knowledge (Fleming et al "Metacognition..." 2012), selection of strategies for memory, learning, life-span approaches (Marković et al, 2021; Shimamura, 2000), learning higher-level objectives (Timmermans et al, 2012), trading off between exploration and exploiting existing knowledge (Marković et al, 2021), balancing effort vs benefits of possible behaviors (Carruthers & Williams, 2022; Marković et al, 2021; Peters, 2010; Fernandez-Duque, 2000), planning (Marković et al, 2021; Cleeremens, 2020; Fernandez-Duque, 2000), monitoring and predicting first-order dynamics (Cleeremens, 2020; Fleming et al "Metacognition..." 2012; Timmermans et al, 2012; Cleeremans et al, 2007; Peters, 2010), control of attention (Whitmarsh, Oostenveld, Almeida & Lundqvist, 2017; Shimamura, 2000), control over working-memory (Whitmarsh, Oostenveld, Almeida & Lundqvist, 2017; Shimamura, 2000), internal conflict resolution (Shimamura, 2000; Fernandez-Duque, 2000), maintenance of cognitive homeostatic needs (Peters, 2010; Shimamura, 2000), emotion regulation (Shimamura, 2000), theory of mind (Carruthers & Williams, 2022; Cleeremens, 2020), and in support of social cooperation by enabling a group to identify the individual who is most certain about some decision point (Cleeremens, 2020; Fleming et al "Metacognition..." 2012; Fleming et al "Prefrontal..." 2012; Cleeremans et al, 2007).

Meta-cognition can be viewed as having a few aspects:

* meta-representation ...explain...
* meta-control (observation only vs control) ...explain...
* first-order vs conscious processes ...explain...

Some running questions have cropped up out of those studies:

* To what extent does meta-cognition actually need meta-representations? ...examples...
* To what extent are meta-cognitive processes truly conscious? Or are they just first-order processes that influence verbal report without direct conscious access? ...examples...
* axis:
  + with/without meta-representation
  + first-order only vs higher-order network architectures
  + extent to which different human meta-cognitive behaviours employ meta-management.
  + relationship to conscious experience.
  + caught up in questions about whether consciousness has any functional purpose (Rosenthal, etc)

# Computational Theories of Consciousness

*todo -* Define: computational.

Computational theories of consciousness promise to provide the mechanisms underlying consciousness, and this should hopefully cover some of the processes of meta-cognition. For example, Global Workspace Theory (GWT) posits that groups of functionally specialized processes cooperate to boost their collective signal strength and thus gain the right to broadcast to all other processes within the system, via the so-called *global workspace*. Stable collaborations between such processes form contextual *frames*, which influence the behaviors of other processes. Thus, changing external circumstances can be quickly adapted to by changing the set of collaborating processes.

However, GWT is described at a very high level. Furthermore, it fails to develop the mechanisms needed to ensure that the system as a whole is stable, and how an agent built on the theory would learn its objectives and act towards those objectives. GWT, and all other computational theories of consciousness, define a system having internal state. That internal state must be managed somehow as it interacts with perceptions of external state, governs the processes operating within the system, and becomes updated as a result of those perceptions and processes. Many computational theories of consciousness fail to cover this area at all. Others that do focus on the state management processes (eg: bayesian models of consciousness, discussed later) either fail to link back to adaptive flexibility, or do so only in a high level manner.

Thus, the studies of meta-cognition and computational theories of consciousness would both benefit from a more in-depth investigation of the processes and mechanisms for management of the changing internal state that influences perceptions and actions. This is the area of *meta-management* (the more general, non-biological, equivalent of meta-cognition).

The focused study of meta-management will i) identify the problem spaces that require explicit meta-management, ii) elucidate mechanisms of meta-management, iii) help to resolve some of the confusions in meta-cognitive research, and iv) add to the growing body of techniques that are useful in development of artificial intelligence (AI) systems.

This paper attempts to untangle the meta-cognition research confusion by relating biological meta-cognitive needs to the need for meta-management processes within artificial computational models. An argument is presented for the need of specific adaptive abilities within computational models, and for the meta-management processes that can underlie those adaptive abilities.

GWT

HOT

Bayesian Models

Bayesian models view much of brain operation as a system for predicting latent state, and for predicting actions that move latent state towards a preferred latent state. Thus conscious perception is inferred latent state.

This provides a useful backbone to meta-cognition, as inference over brain state.

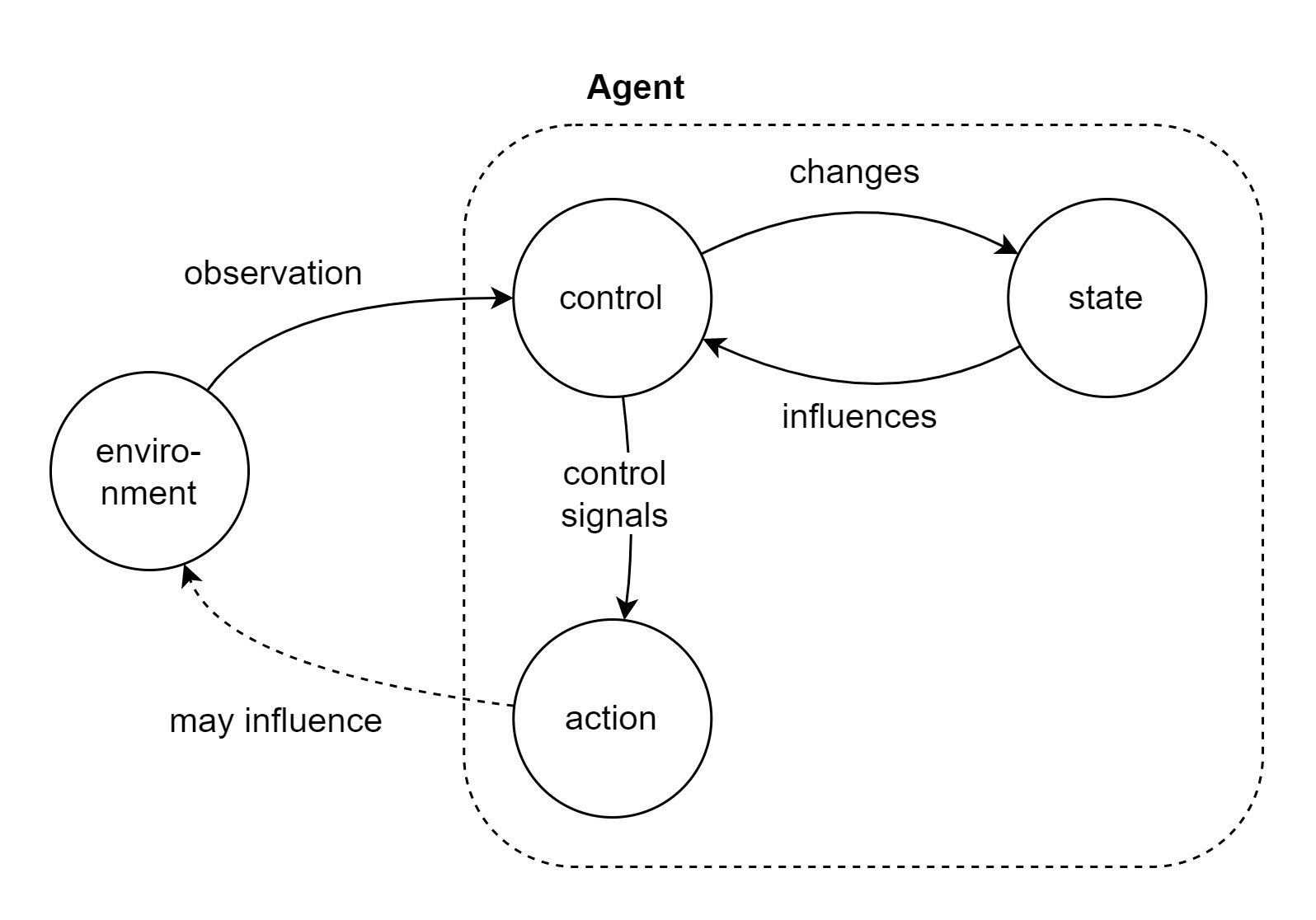
IIT?

# Artificial Intelligence

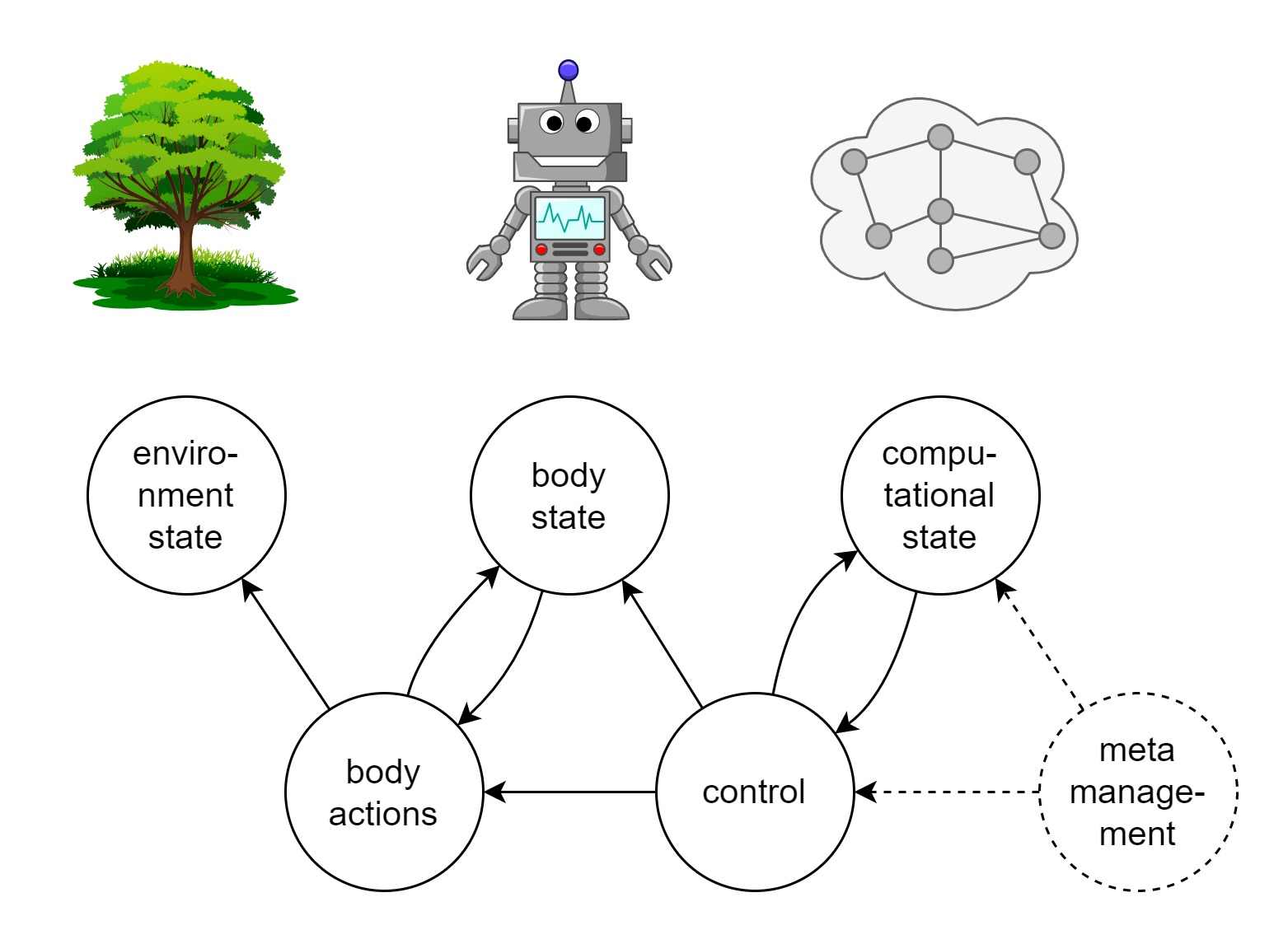
....????....

# State in Embodied Agents

Most computational theories of consciousness indirectly describe the brain as a *state machine*. This is meant here in the sense that the brain has a dynamical state that persists through time, and which influences and is influenced by the cognitive processes. In brains, this includes high-level examples such as knowledge, memories, and life-style choices, down to low-level examples such as messages passed in recurrent loops, levels of arousal, and neuroplasticity.

Figure 2: State machine

As highlighted in Figure 2, the point is that every observation of and reaction to the environment results in a change to the agental system; and that this change affects the agent's subsequent reactions. This broad conception of state machine will be re-used throughout the rest of this paper. Additionally, sometime it will be referred to as a single state, other times it will be convenient to refer to the fact that a system's state is made up of many different components (eg:...) and thus will be referred to in the plural "states".

Figure 3: Three states to manage

Thus, a computational control process can have state that is independent of the state of its external environment. As illustrated in Figure 3, embodied agents with state machine computational control processes have three distinct states: i) the state of their external environment, ii) the state of their (biological or artificial, physical or virtual) bodies, and iii) their internal computational state.

An agent that exists within an environment must monitor and predict the state of that environment. It may also act with the intent to change the environment (eg: put a plate on the table, or lift the object held by the robotic claw).

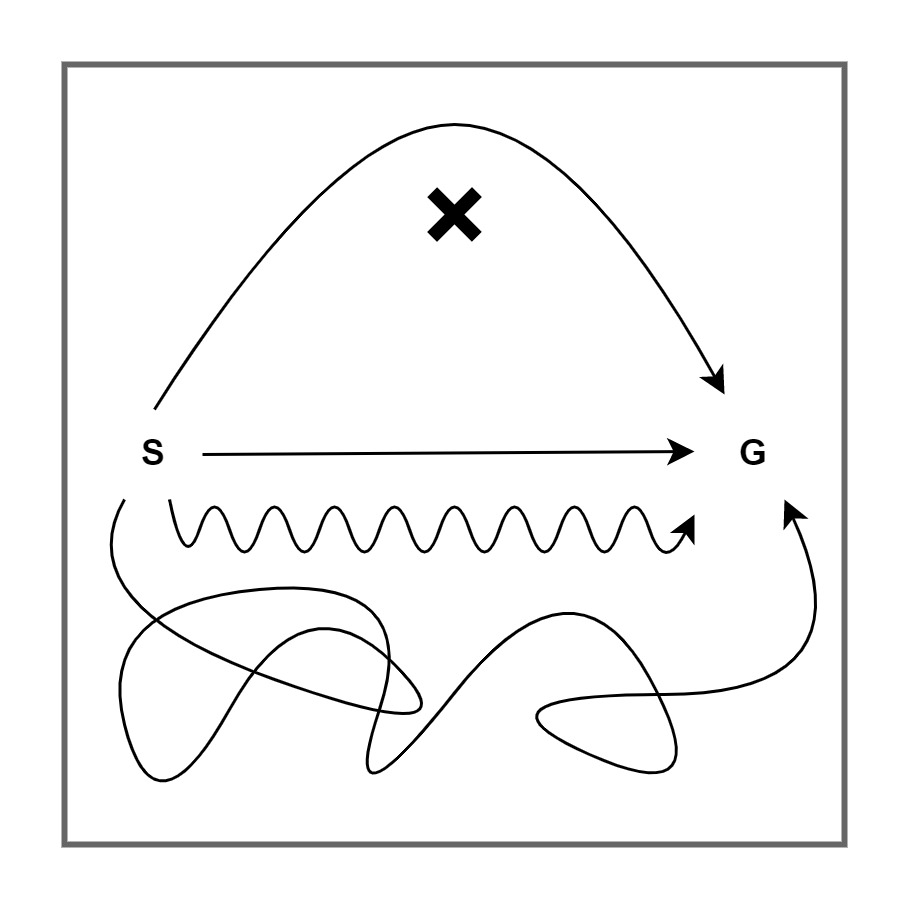
These actions are performed by the agent's body, which itself can be said to be in some state at any point in time. A significant component of the computational control processes are required to monitor, predict, and to tune the body's static state (eg: it's current location and energy levels) and dynamic state (eg: speed and acceleration of arm movement, adapting to resistance in movement due to detritus in gears).

The same can be said for the state of the computational control process itself. But the case for this needs a little more explanation.

## State Trajectories

The course taken by an agent to get from a past state to its current state is its *state trajectory*. Analogous to the path taken by an agent while walking through a maze, the state trajectory describes the path of the agent through state space. Here the state space can refer to its possible locations in physical space, such as in the maze example, or to more abstract possible states, such as an encapsulation of all measurable aspects of the agent's body parts. This provides a useful abstraction away from the low-level details of individual actions. It is a useful abstraction to us for conceptualizing about the agent's behavior. It is also a useful abstraction for the agent itself, as will be seen later.

Not all state trajectories are good ones. Figure 4 illustrates a number of possible state trajectories from start state S to goal state G, while avoiding obstacle X. Each trajectory successfully reaches the goal, but they vary in other ways that may have significant impact to the agent. They length of the trajectory may indicate energy efficiency, which is important for an agent with limited energy reserves. The length may also indicate the time taken, which impacts whether or not the goal is reached "in time". The smoothness of the trajectory can be important. A jagged trajectory might indicate that the agent's physical body is moved in a chaotic way with abrupt stops and starts, causing damage to delicate moving parts from the stresses of that chaotic movement. A smoother trajectory may be easier for the agent to subsequently learn from and reason about in order to improve its later attempts; whereas a more chaotic path may add so much noise to the observations of the trajectory that the agent is unable to detect the most important patterns for such learning.

Figure 4: Good and bad state trajectories

Thus, it is not sufficient only for an agent to achieve its goals, it must achieve those goals through "good" state trajectories – to whatever extent "good" means in the context. This is obviously true for the physical state of the body of an embodied agent. In certain circumstances it is also true for the computational state of the control system. A number of examples are presented next.

## Computational State Trajectory during Body Action

Actions by an embodied agent occur over time. During the time it takes for an agent to move its arm through space from the arm's initial position to target position, the agent will make many observations about the environment and body states. The agent's goal and action plan must be relatively persistent during that time. Otherwise the agent's behavior will be chaotic, with rapid goal and action changes.

Thus, while the agent manages (controls) the trajectory of its body state through the use of its computational state (eg: the given goal and action-plan at the time), it must also *meta-manage* the trajectory of that computational state. In this case, the agent's computational state must to some extent resist change influenced by new observations.

## State Trajectory during Multi-step Processing

Not all actions can be decided upon immediately. Any computational system has a limit on its bandwidth: the level of complexity of computation that it can perform in a single pass from input to output. In the field of Artificial Intelligence, deep neural networks use many layers (sometimes hundreds) to improve that bandwidth (citations). Recent work (citation, "loops are the way forward") has found that deep neural networks can be replaced by shallower networks that employ end-to-end recurrency (where top-level output is used as feedback into the bottom-level input layers). These shallower *macro-recurrent* (..definition...) networks provide the same or better performance, but have less free parameters and are faster to train.

Thus, such an agent can execute a trajectory through computational state space, without performing any body actions. And this state space trajectory needs to be managed just the same as above. Just as in the above, in order to maintain stability the agent needs to i) observe the state space trajectory, ii) apply some objective measure to decide upon the relative effectiveness of the trajectory, and iii) act to change the trajectory if a better one is available.

A number of examples of potential "control problems" in such a system with "deliberative" capabilities were given by Beaudoin (1994):

* **Oscillation between decisions**. Wasteful re-assessments of decision points, leading to a meta-stable (oscillating) but stagnant (ultimately achieving nothing useful) state.
* **Insistent goal disruption.** Repeatedly getting distracted by competing goals that have been previously disregarded.
* **High busyness.** Attempting to multi-task between too many goals, leading to poor outcomes.
* **Digressions.** Choosing to deliberate over some sub-goal, and then loosing track of the "big picture" by forgetting to return to the overarching goal.
* **Maundering.** Getting stuck deliberating over the details of a goal without making a decision.

## State Trajectory during Iterative Inference

A third example of the need for control of state trajectories is found in the case of *iterative inference*.

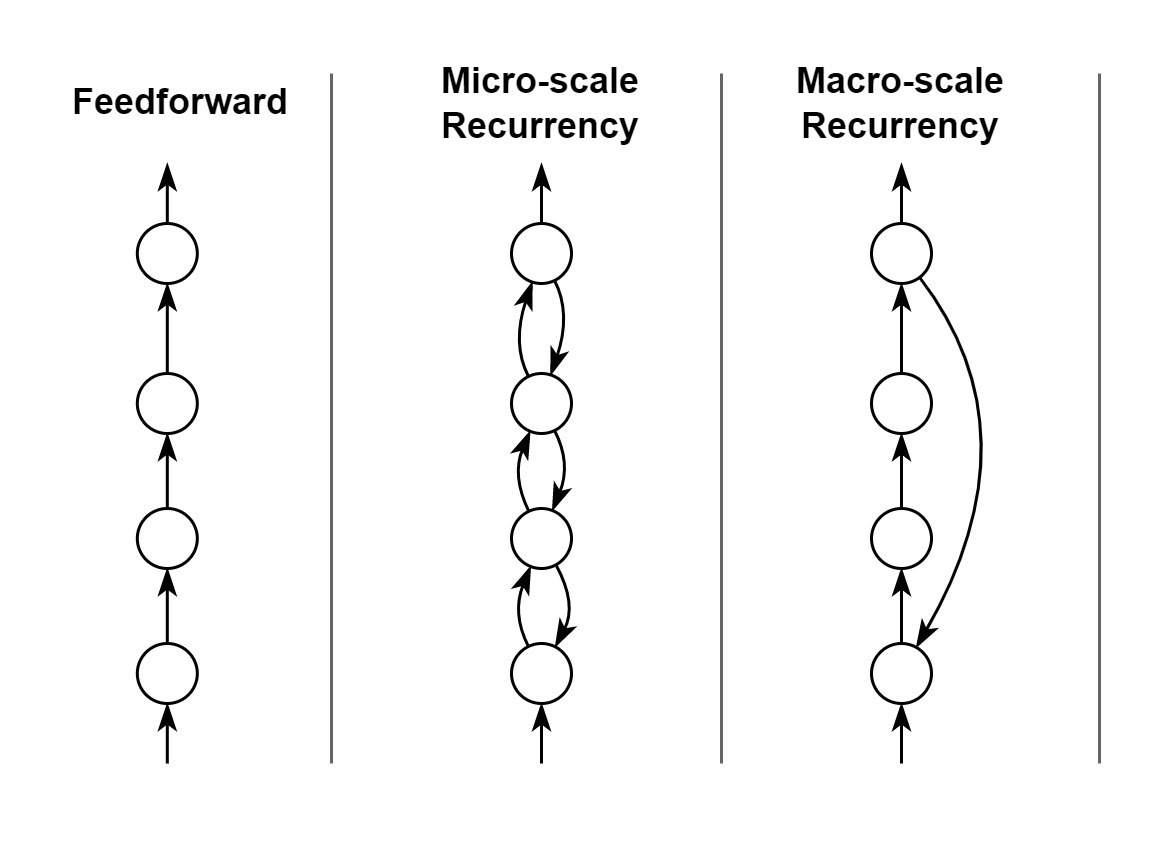
*todo Here an ............*

This is similar to the above examples of state trajectories. The computational state needs to be relatively persistent during the execution of the iterative inference process, as part of that computational state reflects the context of the inference. In other respects, the computational state is part of that iterative process, and thus changes with it. The trajectory of the changing computational state needs to be managed.

## Recurrency

*todo - I might not need this "Recurrency section". Really the point about state trajectories during iterative inference is all that this "Recurrency" section is trying to get at. So maybe I don't need to go into such detail and can just explain it above in simple terms.*

Feedforward - has no state.

Figure 5: Types of recurrency

Micro-scale recurrency - iterative predictive networks; hierarchical architectures.

Macro-scale recurrency - state machine loops; large-scale recurrency in cortico-thalamic system and others.

## Section Summary

***todo*** - Finally, we have three areas in which state trajectory must be managed:

* during iterative prediction (micro-scale recurrency)
* during loop execution (macro-scale recurrency)
* while waiting for actions to play out

# Meta-management

*todo ... with all that is said, it is now time to formally define meta-management:*

* *Cleeremans defines meta-management as not having causal effect on the outcome of the first-order network. This suggests an observation-only role for meta-management, which limits its usefulness, and elicits ideas of epiphenomenalism.*
* *I suggest that meta-management always also includes active control of the first-order network, and thus is causal, but only in a specific way.*
* *Possibly better: meta-management is a higher-order process that monitors and controls the first-order network in such a way as the first-order network has no observability of the existence of operation of the higher-order meta-management network. => However, even this fails because it conflicts with "integrated meta-management".*
* *Another: meta-management is how a system maintains (the consistency, stability, convergence of) its own internal state, while the system manages its interactions with its external environment.*
* *Thus, meta-management may take many forms - and the purpose of this paper is to elucidate on those forms and the implications thereof.*

The following four chapters look at meta-management and its options from different perspectives.

# Meta-management Needs

Why might we need to add meta-management processes to connectionist architectures? Deep AI techniques have had many successes of late (citation). However, these networks still lack some of the most basic adaptive capabilities that we see in many biological organisms (citations, eg: sloman).

Here some specific meta-management features are discussed in the context of how they might improve connectionist computational systems such as deep AI architectures. This "design stance" is useful as a means for teasing out the lower level mechanisms that may underlie much higher-order behaviors such as meta-cognition.

*Todo: discuss each of the following in terms of:*

* *benefits*
* *relationship to meta-representation*
* *relationship to observation/logical reasoning/control components of meta-management*
* *theoretical mechanisms*
* *empirical evidence, if any*
* *give citations of AI examples showing the benefits of each of these*

## State trajectory control

As discussed in detail in an earlier chapter, there is a strong case for the need to actively manage the trajectory of the agent's computational state. Three contexts have been highlighted for this need:

* during iterative prediction (micro-scale recurrency)
* during looping multi-step execution (macro-scale recurrency)
* while waiting for actions to play out.

Mechanisms underlying state trajectory control can include:

* Observing performance over time
* Predicting future outcomes from current trajectory
* Predicting expected future utility of current trajectory, and comparing against that of other predicted possible trajectories.
* Applying tuning control where current trajectory is sub-optimal.

Meta-management of computational state is not necessary in all computational systems. Many of the most successful deep AI systems today undergo a training phase, where externally controlled learning pressures are applied (eg: supervised learning, re-enforcement learning), followed by a non-learning runtime phase. In these, the state trajectories are effectively pre-configured during the training phase. Some contexts in which active self-management of computational state trajectories include:

* Agents with continuous and/or online learning
* Hierarchical architectures. Agents with a separation between higher- and lower-order goals and control systems, whereby the higher-order control systems apply context or control over the lower-order control systems effectively employ meta-management.

## Objective learning

How does a continuously learning embodied agent know which actions are better than others? This decision is tied to the agent's *objective*: it's ultimate goal that influences all other goals. For example, to eat and stay healthy in order to survive. Or, to produce as many staples as possible in as little time as possible (citation). If the agent is not pre-configured with its objective, then it must learn that objective.

An agent in the human world requires the use of inedible metal tokens (coins), which are used in complex ways for the purpose of life preservation. The involvement of such an inedible metal token as part of some process (eg: doing a job and being payed) does not necessarily immediately result in a life sustaining outcome. Thus, without any other information, it is hard for the agent to learn the relationship between that inedible metal token, the processes that it must be involved in, and the life sustaining result. This is known in the AI community as "sparse feedback", and it poses a particularly difficult problem for continuously learning agents (*citation needed*).

Another problem for a continuously learning agent is known as the "exploration-exploitation dilemma" (*citation needed*). The agent gains knowledge about its world and itself by exploring places, things, and behaviors that it knows little about. When the agent needs to achieve a goal, it may know that it can achieve the goal via its existing knowledge (exploitation), but it may be able to achieve that goal in some better way if it were to explore more first; it also may not. The dilemma concerns how the agent chooses between exploration and exploitation at any given moment.

Sparse feedback and the exploration-exploitation dilemma make objective learning difficult. One solution is for the agent to build simplified models of its environment, itself, the behaviors it can perform, and how those behaviors influence different outcomes. This becomes the agent's "knowledge", and somewhere within that knowledge a continuously learning agent builds a structure that ultimately governs its behaviors and goals – that is, an objective that it infers over time.

Importantly, those models can have different forms, and their forms influence what kinds of inferences the agent can draw from the knowledge, and consequently how they can be used for other management and meta-management purposes. A discussion of different models is presented in a later section.

## Mode selection

strategy / goal / context / module / attention selection

Exploration vs exploitation

Priors or other tuning mechanism. Even priors on meta mgtmt layers.

eg: adjusting priors, inhibition, excitation

## Mode identification

eg: meta-strategy clustering.

Identification of component/process interactions and effects in order that they may be chosen via mode-selection. Requires observation of systems in such a way that makes it possible to differentiate their parts and/or processes. Thus needs model access.

## Distributed cooperation

Managing competition and cooperation between many sub-processes.

In humans, probably a first-order network concern because we don't consciously experience and control that process. But it is in principle possible to be done at either level.

eg: adjusting priors, inhibition, excitation

## Certainty measurement / reaction

Eg: low level simulations linking certainty encoding to attention.

Not sure how used for meta mgtmt, but has a plausible low level mechanism.

# State Representation

## Dimensionality reduction

Meta-management is management over reduced dimensionality. This suggests re-representation.

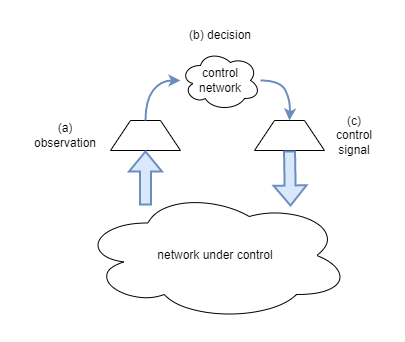


Figure 6: Dimensionality reduction. A large network under control (NUC) can be observed and controlled efficiently by a much smaller network. Dimensionality reduction benefits the process in three ways. (a) a reduced dimensionality observation of the state of the NUC enables the control network to interpret that state without an exponential increase in the total number of neurons in the system. (b) When operating over a reduced dimensionality, the control network can learn with fewer training iterations, and apply more advanced decision rules with less resources. (c) the output of the control network can also be in a dimensionally reduced space, further simplifying its computations, as hierarchical models provide a mechanism for low-dimensionality signals to control higher-dimensionality networks.

**Observation:** A network cannot micro-manage itself. In order to observe the full state of every neuron would require at least just as many neurons again, or probably many times more. Thus, the dimensionality of the observation of system state must be significantly reduced for the practical purpose of avoiding an exponential scaling out in the number of neurons of the total system. Predictive mechanisms are well suited to this. Typically predictive mechanisms are used to infer the hidden *latent* state of a system, based on observations obtained about that system. A side effect is that the inferred latent state is only an estimated representation of the true system latent state, and consequently it usually has significantly less dimensionality than the true latent state. Thus, the predictive mechanism can also be seen as a dimensionality reduction mechanism that produces a self-stabilising (auto-convergent) simpler representation of the state of the system under observation.

**Decision:** A reduced dimension state space is beneficial for the control logic. Learning good control methods/parameters is more efficient and more stable in a lower dimensional state space. Additionally, the control system can apply more complex rules with less resources than it would otherwise need.

**Control:** Lastly, a reduced dimensionality is also good for the final output of the control system, for all the same reasons as above. However, that reduced dimensionality may need to be subsequently up-scaled if it is to control at the low-level scales. Thankfully there is well-established precedent for that in the form of U-Nets (citation) and in hierarchical predictive models (citations).

Most of the other "why"s mentioned here benefit hugely from using higher-order representations because it reduces the dimensionality of state spaces for: monitoring current internal state, monitoring external feedback, learning associations. Additionally, where those higher-order representations are inferences over the latent states, then they unify multiple sources of information (different sensory modalities, information presented over time).

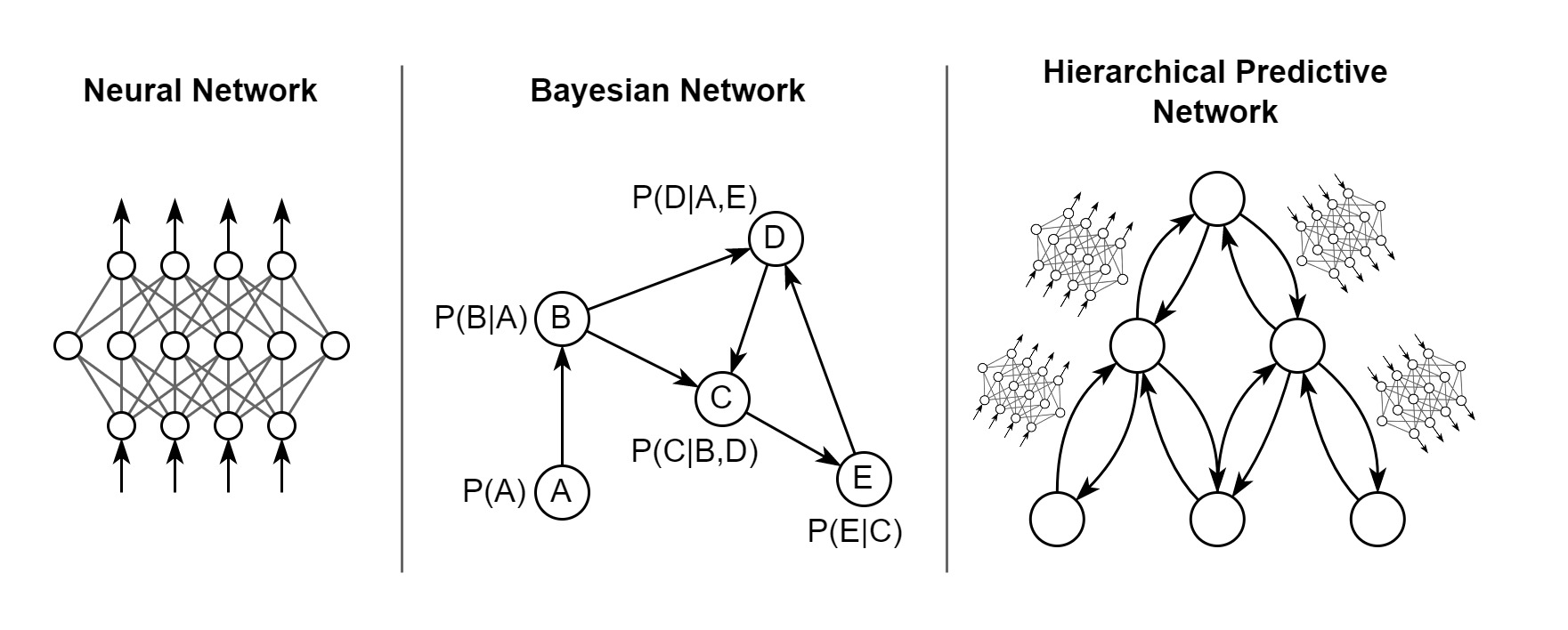
# Model Structure

## Good Regulators

....good regulators need to be a model....complex regulators need to have a model...

According to the *good regulator theorem*, if the agent is to regulate the environment state it must be a "model of the system" (Conant & Ashby, 1970). Furthermore, we can say that the efficiency of the agent to regulate its environment depends on its accuracy in modeling the system. Errors in the accuracy of the model result in errors in the regulation of the system. In learning agents, those errors are used for subsequent training of the model.

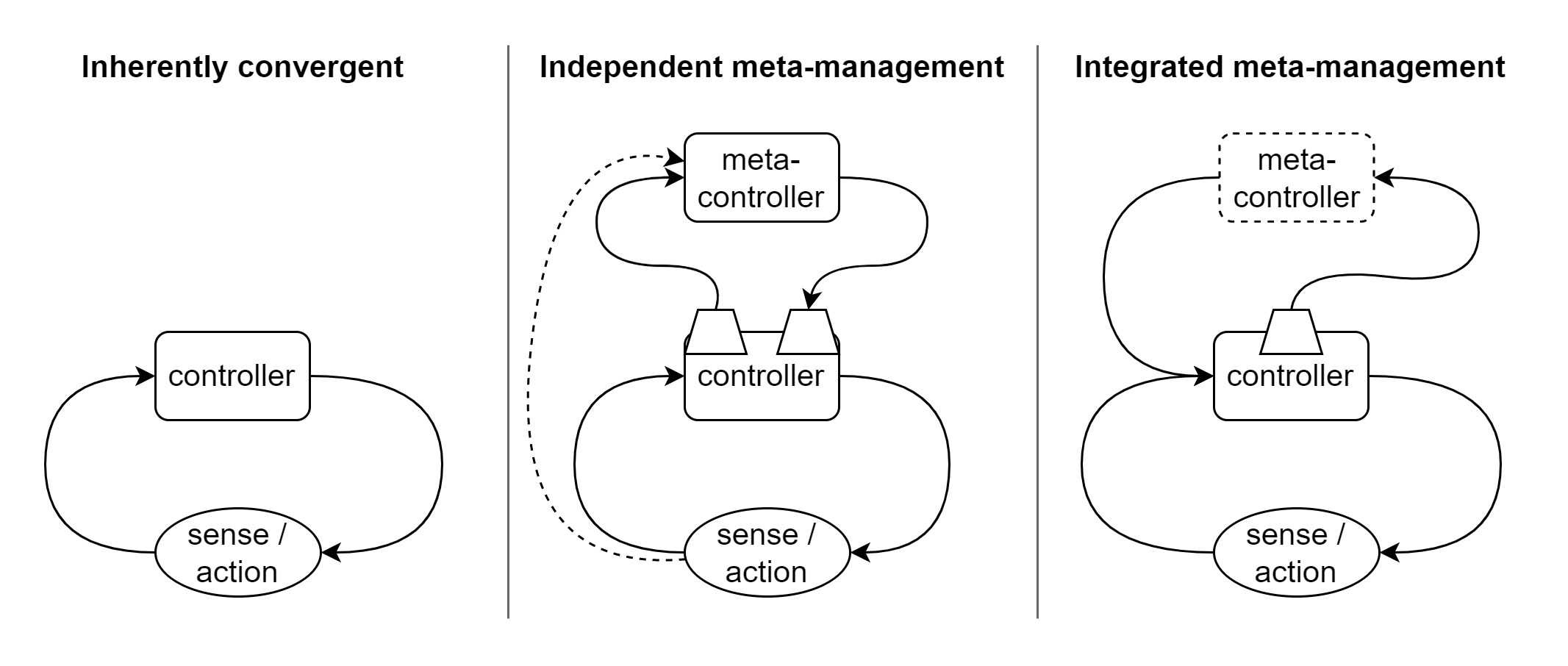
## Kinds of Model Representation

Figure 7: Forms of of model

# Meta-management architectures

(for each section, explain what it is, how it might be implemented, and existing papers)

Integration styles: what options are available for building a stable system?

Figure 8: Meta-management architectures

## Implicit meta mgmt / Inherent-convergence / first-order (meta-)management

* This is the null-hypothesis architecture - one which doesn't need meta-management because it's inherently stable.
* Examples: engineered solutions, static systems, predictive models.
* Con: presumably less rapid adaptation as it's always running under the same (implicit) objective measure. Presumably, the advantage of explicit meta-mgmt is a higher-order effect: the main systems operate against the first-order objective; the first-order objective is customised by meta-mgmt; the meta-mgmt system operates against a higher-order objective. By working against a simplified representation, and thus a heavily dimensionally reduced state space, it is more efficient to train.
* citations - suggestions that this is suitable:
  + Peter Carruthers, David M. Williams (2022). Model-free metacognition.
  + There was another paper asking whether meta-cognition is needed for consciousness, but I've lost it.

todo: Predictive mechanisms are inherently convergent.

## Independent meta mgmt

* Totally independent. First order network gets no feedback from second order network. But second order network tunes first order. Output of first order never represents anything from 2nd order network.
* Examples: hierachical models could be considered a case of this, though they are perhaps arguably only a simple form?
* citations
  + Most synthetic models assume this architecture.
  + eg: Cleeremans.

## Integrated meta mgmt

* Pro: domain dependent knowledge and processing
  + (search meta-management note for source) "Karmiloff and Clark's re-representation theory is primarily focused on re-representation of knowledge of external environment. This re-representation requires a relatively complex system to perform its functioning. But re-representation can just as easily be useful in development models of one's own mind, and thus this requires access to that same capability."
* Con: less stable, but presumably there are mechanisms. Eg: active inference.
* Specific case: Observation + objective measure only
  + Con with other meta-managements is that they too need to be trained, so what objective do they train against?
  + This architecture has the meta-mgmt loop as just a meta-awareness with no active role. It just mirrors the brain's state back onto itself, along with a judgement about the efficacy of the current trajectory.
  + The brain's unconscious processes can then use that for meta-mgmt, for example under an Active Inference model.
* Specific case: Observation only
  + It's possible that even the judgement too can be done via the main loop's unconscious processors, particularly under an Active Inference (inherently-convergent) model.
  + I find this the most compelling architecture.

How? inherently convergent (predictive mechanisms).

# Control: How?

* priors
* attention - how?
* inhibition / excitation

# Meta-management in the Human Brain

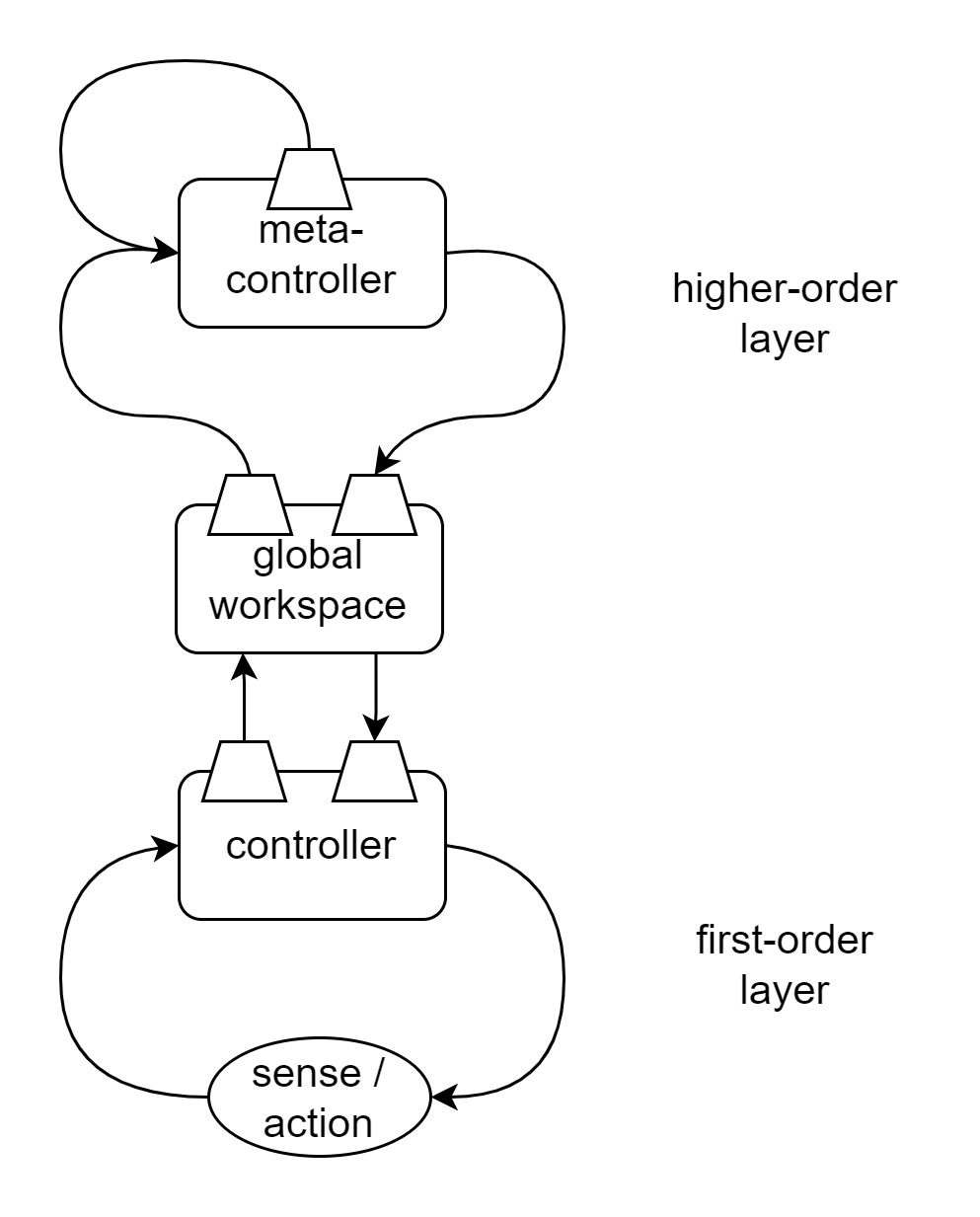
This paper would not be complete without drawing some speculations about the nature of the meta-management architecture (or architectures) within the human brain, and an attempt to link that back meta-cognition and human perceptual experience in general.

## A Speculative Human Meta-management Architecture

First, some observations:

* The brain is tremendously complex so it stands to reason that the brain likely employs multiple different meta-management systems, operating at different levels.
* It is well confirmed that we have experiential awareness of only a small fraction of the full state of the brain (citations). Thus it seems reasonable to conclude that the conscious part of the brain (the part that has experiential awareness of things) receives only a dimensionally reduced representation of the brain state.
* Unconscious automatized processes can do a lot on their own without conscious involvement (Baars, Rosenthal).
* One significant benefit of automatized processes seems to be that it frees up higher-order processes for other things, such as contemplating longer-term issues. Thus the brain is able to operate automatized processes while cogitating on completely unrelated things, all in parallel. We see this is the form of the "default mode" network, and mind-wandering.
* Experiential awareness seems to be very much about observation of perceptions of the external environment combined with perceptions of one's own state. So, presumably, anything that is conscious requires a meta-management feedback loop.

With those observations in mind, a speculative meta-management architecture of the human brain is as follows (illustrated in Figure 9).

Figure 9: Speculative human meta-management architecture

It has (at least) two hierarchical layers: a first-order layer and a meta-management layer. The first-order layer receives, processes, and controls first-order low-level signals. It employs learning mechanisms, such that sufficiently well learned (automatized) processes can be executed without further involvement from the meta-management layer. For a static or well-rehearsed context, an automatized process is effectively inherently convergent, and does not need meta-management. This is especially true where the underlying mechanisms are predictive.

The meta-management layer provides an independent meta-management architecture for the first-order layer. The meta-management layer constantly predicts the expected outcome of the first-order layer, in the same way that our sensorimotor system constantly predicts the expected outcome of our actions (citation). When the first-order layer performs as expected, the meta-management layer does not get involved. It does get involved, however, when its prediction of the outcome from the first-order layer's behavior is either different to what it's currently observing, or because it predicts an undesirable outcome. This may, for example, also occur where context in which the first-order process is operating is unusual. When necessary, the meta-management layer influences the first-order network through changes to priors, attention, etc.

The meta-management layer can also do computational processing for its own purposes, independent of the first-order layer. The reason for this is that it employs a number of complex systems, including modeling and deliberative systems, that are necessary for the control of the first-order layer; however, when the meta-management system is not controlling the first-order layer, those systems are free to be used for other things.

With such a complex and adaptive meta-management system, it too needs meta-management. This is solved by the meta-management layer further employing an integrated meta-management architecture: it meta-manages itself.

The interaction between the two layers is via a global workspace. The content of that global workspace can be influenced by either layer, with different aspects of the state within the global workspace having different influences on the processing happening within the two layers. And this influence changes over time as the global workspace state changes. In this way there is a dynamically changing degree of interaction between first-order and meta-management layers, and the meta-management layers can leverage the domain-specific capabilities of the first-order network where it needs to.

The global workspace holds a dimensionally reduced representation of the state of the first-order network. Thus the meta-management layer needs only build simplified models for the prediction and control of first-order processes.

## Meta-management and Meta-cognition

The above architecture fits with some observations from behavioral studies, and offers some possible resolutions for remaining contentious issues.

... todo ...

Most meta-cognitive behaviors could be achieved through first-order means, without the need for meta-cognition. This is consistent with the idea that automatized behaviors are first-order processes that don't need constant meta-management involvement.

Whether consciousness has any utility? Anecdotally, experiential awareness seems to coincide with explicit meta-management processes. This would seem to have something to do with the self-observation component of the integrated meta-management architecture. That self-observation component definitely has utility – without it the entire system would become unstable. Thus, assuming that there is something about the self-observation feedback loop that causes the effect of experiential awareness, and assuming that the self-observation feedback loop and experiential awareness are somehow intimately linked (ie: one cannot exist without the other), then it can be said that consciousness has utility.

...todo...sholud be able to link back to the specific "recurring problems" with meta-cognitive research.

# Summary

More empirical studies needed.

* It is worth pointing out that, on the face of the evidence presented so far, all of the architectures described above are equally effective. Even the higher-order objective theory of meta-mgmt could be implemented in an inherently convergent architecture using hierarchical solutions (eg: hierarchical active inference). But, as these areas go, further research will likely find the niches where each architecture are better than the other. Obviously empirical results are needed here.
  + eg: Metcalfe’s CHARM model (Metcalfe, 1993). See reference in note: Meta-management > Shimamura  2000, "Toward a Cognitive Neuroscience of Metacognition"
  + eg: Kimberg and Farah (1993). See reference in note: Meta-management > Shimamura  2000, "Toward a Cognitive Neuroscience of Metacognition"

# Introduction

Computational and connectionist theories of consciousness are inherently built upon the idea of a state machine and loops, but they fail to draw specific reference to this dependency.....GWT, etc. etc. (citations). Thus, there is an avenue for further insights.

The field of meta-cognition has begun to make inroads. Originally focused on the most outward behavioural aspects of the fact that people who are more aware of their own learning strategies, strengths, and weakneses, do better. Now, meta-cognition research investigates how the brain performs those behaviours. Furthermore, many have suggetsed that meta-cognition may be the basis for consciousness itself (citations). Meta-cognition has been implicated in ............(behaviours, with citations)....

Like theories of consciousness, much of the meta-cognition theorising is at the level of behaviours or whole of brain processes. Some attempt to draw references to specific brain regions (..citations..) but that work is still very speculative. Only a few ...(citations)... have attempted to simulate such processes in connectionist models. Those simulations are usually very simple. For example, they (...citations...) simulate the construction of higher-order representations about certainty, but don't use that as a feedback signal for the system to incorporate into its processing.

The present paper attempts to strengthen the meta-cognition research in two ways. First, it attempts to bridge the gap between existing connectionist mechanisms and meta-cognitive theories by highlighting specific low-level connectionist mechanisms that might form the basis for meta-cognition. Secondly, it examines different connectionist architectures, and shows how those architectures lead to different observable results.

In order to focus on practicality, a "design stance" is taken.... (citation and explanation).... This leads us to focus on the bottom-up design, which serves two purposes. i) It provides a stronger proof of the value in the arguments, and ii) it offers direction for using the knowledge to build systems with these capabilities.

# Agents as Regulators

# Meta-management Architectures

# Conclusions

Bottom-up design stance has been taken. This offers opportunities to build systems based on these architectures, and to measure empirically their relative benefits for different problem domains.

# Regulation

An autonomous embodied agent, depending on its purpose, may need to control either its environment, itself, or both, towards some static or dynamically determined target. That agent can be described as a *regulator* of its target system.

For example, an agent that regulates its environment operates within a system containing environment state *senv* which changes with some ambient dynamics *denv(t)*. The agent must perform an action, *aenv*, against the environment in order to regulate itself towards some target. After an action has been executed the environment state outcome *oenv* is influenced by both *denv(t)* and *a*env. This can be summarized as the following equation:

According to the *good regulator theorem*, if the agent is to regulate the environment state it must be a "model of the system" (Conant & Ashby, 1970). Furthermore, we can say that the efficiency of the agent to regulate its environment depends on its accuracy in modeling the system. Errors in the accuracy of the model result in errors in the regulation of the system. In learning agents, those errors are used for subsequent training of the model.

An embodied agent with complex actions may require an additional level of regulation. For example, an animal must not only regulate its external environment, but also regulate its own physical state. This includes both maintaining homeostasis and controlling the efficiency and effectiveness of its actions against the environment. The agent performs action *abody* against its body with the intent to regulate the body towards efficiently achieving environment action *aenv* while satisfying its requirement for body homeostasis. Such an agent thus operates in a system that additionally has body state *sbody* with ambient dynamics *dbody(t)*. The agent performs action *abody* against its body, producing outcome *obody*, summarized as follows:

Agents that incorporate multi-step processing have a third kind of action: one that changes its internal data state without affecting its physical state. This system requires regulation for the same reasons as for environment and body, but such *non-physical* actions do not elicit any change to *sbody* or *senv*. Thus the agent must regulate its non-physical state *smind*, having ambient dynamics *dmind(t)*. The target state in this case is dynamically inferred based on its requirement for environment action *aenv*, body action *abody*, and possibly for some form of non-physical homeostasis of *smind*. In order to regulate towards that target it performs action *amind* producing outcome *omind*, summarized as follows:

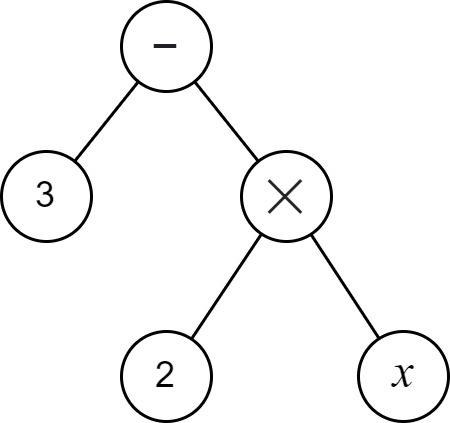
By way of example of the importance of such mind regulation, consider the case of fluent aphasia caused by damage to the Wernicke's area[[1]](#footnote-2) of the brain. Individuals with fluent aphasia can easily produce speech, but it is typically full of many meaningless words and often unnecessarily long winded. Wernicke's area is associated with language comprehension. In a neurotypical individual, the comprehension of their own vocalizations provides a corrective mechanism. This illustrates the importance of feedback in the regulation of one's own actions, and by way of analogy extends to the regulation of non-physical actions.

# Models

All of the systems described above are of the form . The production of the optimal action *a* for a given situation can be computed by a function, *f*, such that . In this way, function *f* becomes a *model* of the system in exactly the way meant by Conant and Ashbey. There are many different ways of constructing such a function, with implications on how much its inherent model can be introspected for purposes other than merely computing the next action.

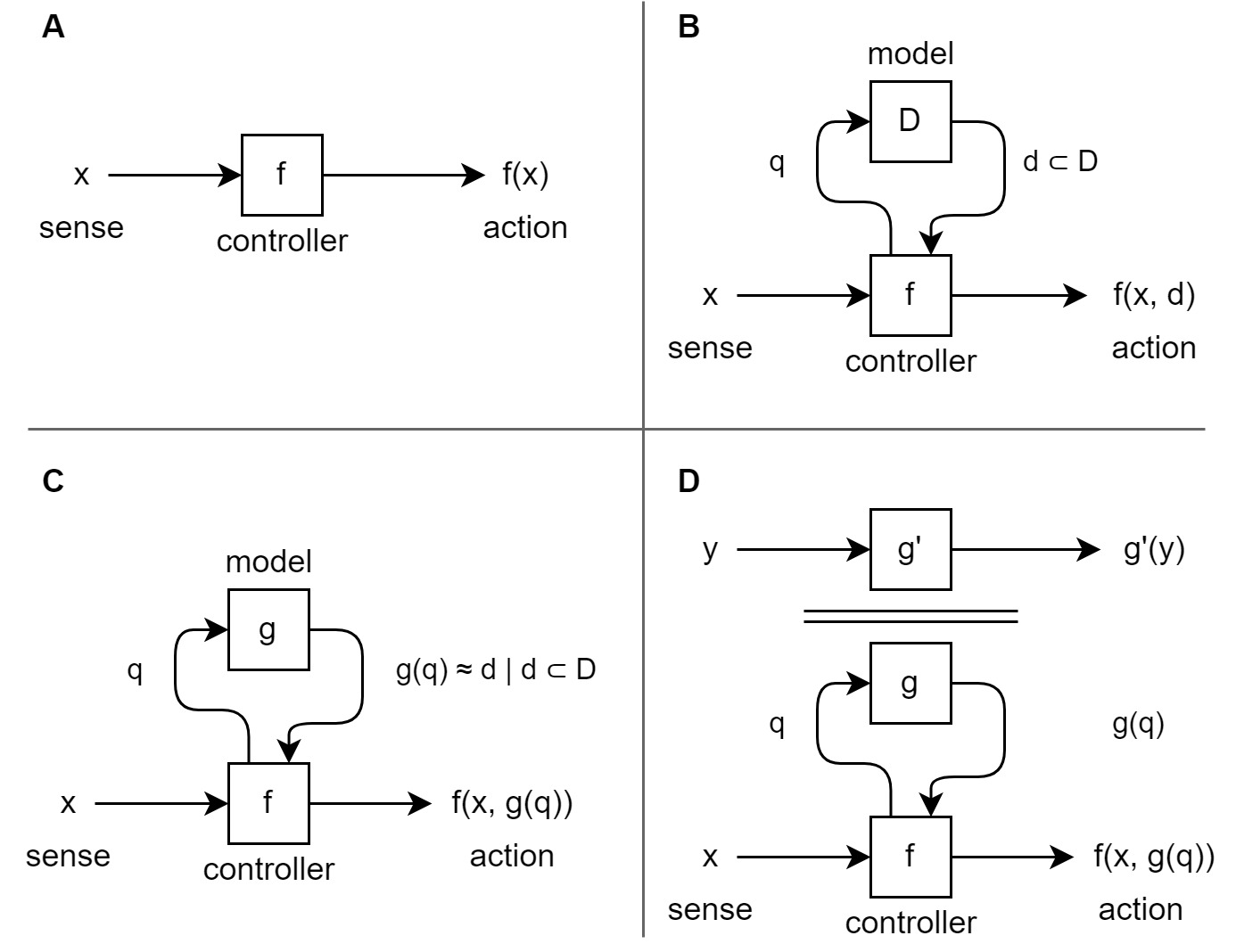
Consider the following function. This function is, for example, effective at predicting the action required to regulate towards a target state of 3 by doubling the input signal and comparing to that target state. However, an agent that merely uses this function to calculate actions cannot inspect anything about the function other than the actions it calculates for different inputs.

Alternatively consider Figure 10, which shows an abstract syntax tree[[2]](#footnote-3) (AST) of the function above, of the sort used by computer science to parse an expression within a software compiler. Instead of using the above function, a regulating agent could use this AST to calculate its next action and achieve the same outcome. However the AST is a more explicit model of the dynamics being regulated. The components of the original function are represented individually and thus they can be individually queried. So here the AST can be introspected and much more can be derived from it that may apply either to the system being modeled or to how the AST models that system.

  
Figure 10: Abstract syntax tree of 3 - 2x

To examine the introspective opportunities further, consider the task of constructing a set, *F*, that contains all beliefs that may be drawn from the model. In the case of the function, pairs of input and output action values are all that can be drawn from the model, ie: <0,3>, <2,-1>, <-3,9>, <-1,5>, etc. The AST supports the ability to draw those same pairs of input and action values. However the AST also supports that many other beliefs may be drawn from the model and added to *F*. For example that i) the target is 3, ii) input signal *x* is significant to the calculation, while *y* and *z* are not, and iii) the execution of the function depends on the operations of *subtraction* and *multiplication*.

So, it is clear that different architectures enable different levels of *introspection* of the underlying models. What about the case for neural networks? In the modern use of artificial neural networks (ANNs), it is commonplace to refer to ANNs as a *function approximator* (Goodfellow et al., 2016), and indeed many networks fall into the category of a function. For example, in *model-free* deep reinforcement learning (RL) an ANN is used to calculate either the next action or the expected value of all possible actions given the current state (Lazaridis, Fachantidis & Vlahavas, 2020). The architecture of the RL algorithm treats the ANN as a function without any introspective capabilities. See Error: Reference source not found(A) for an example. There is also *model-based* RL. One variant of model-based RL, illustrated in Error: Reference source not found(C), uses ANNs to predict the expected outcome of executing an action. The introspective ability here is the same as for model-free deep RL - the ANN is treated as a function. For the RL models mentioned so far, the set *F* of beliefs is of similar content: *F* is the set of <state,action> or <state,action,outcome> tuples. There do exist forms of model-based RL that use something more akin to the AST, usually where there is a known physics model that is represented mathematically, and which could potentially be used to introspect for more than just <state,action,outcome> tuples, such as is illustrated in Error: Reference source not found(B). However, a significant point to note here is that ANNs, and probably neural networks in general, do not lend themselves to introspection on their own.

  
Figure 11: Different architectures for *modeling regulatory actions against the environment.* In (**A)**, the controller determines the next action by executing function *f* against sense input *x*. The function may, for example, be an ANN that is trained through many iterations of an appropriate learning algorithm. Function *f* merely models the best next action without modeling any other aspects of that environment and thus cannot be used to introspect anything other than the next regulatory action. In (**B)**, set *D* holds an explicit model of the environment which can be arbitrarily queried (*q*) to gain insight about any aspect of the environment that is encapsulated within *D*. Controller function *f* uses that to determine the next action. In (**C**), set *D* is replaced by function *g(q)* which approximates queries against *D*. This architecture is commonly used in AI where the dynamics of the environment are too complex to determine a priori, and *g(q)* is built as a second ANN that is trained through exploration. In (**D**), the secondary system *y = g'(y)* models some aspect of the environment other than the next regulatory action. For example, it may observe and predict long term trends in the environment state. Potentially further additional modeling systems are required for each additional aspect of the environment that needs to be modeled.

For that reason, a third form of modeling system exists, whereby a secondary model predicts the behaviors of the former, such as is illustrated in Error: Reference source not found(D). The secondary model may, for example, be a second ANN that captures aspects of the same underlying system but at a more macro level, and it may be more suitable for integration with other data. This macro representation is at the core of the theory of Higher Order Thought Theory (HOTT) (Rosenthal, 1997 & 2006), and of recent theories based on it such as Hierarchical Active Inference (Giovanni et al, 2018) and Integrated World Modeling Theory (IWMT) (Safron, 2020).

# Schemas

The lack of introspective ability of a simple function contrasts with the introspective ability of a human. Psychology has long identified in humans the existence of a model of the individual's body – known as the *body schema*. A good definition is given by Morasso et al (2015):

*"In summary, we view the body schema as a set of fronto-parietal networks that integrate information originating from regions of the body and external space in a way, which is functionally relevant to specific actions performed by different body parts. As such, the body schema is a representation of the body’s spatial properties, including the length of limbs and limb segments, their arrangement, the configuration of the segments in space, and the shape of the body surface".*

So the body schema is a model used in production of action control by integrating information from our main physical senses and the proprioceptive senses (Proske & Gandevia, 2012). That model can also be introspected – for example, we can know where our hands and feet are without seeing them – and those introspections can become the topic of subsequent thought. But there are aspects of the model that cannot be introspected – for example, we have no observability of the arrangement of the sense nerves used to infer the hand and feet positions, or of the effector nerves used to actuate their muscles.

This paper hypothesizes the existence of a second schema, the *mind schema*, that performs an analogous role for the regulation of the mind and non-physical actions. Anecdotally, this seems highly plausible within humans given our introspective ability towards our own mind's capabilities. For example, we can know that we are good at focusing, but struggle with math, that we are more creative when background music is present, and that we need the support of tools to help remember people's names (eg: a notebook). The underlying notion here is that the mind schema helps us to control, monitor, predict, and rationalize about our mental structure and actions in the same way that our body schema does that for our physical structure and actions. It is the regulatory model for our non-physical actions. Additionally, just as for the body schema, there is a distinct delineation between what can be introspected and subsequently thought about, and what cannot.

The suggestion of a mind schema has also been made in the form of *Attention Schema Theory* (Graziano & Kastner, 2011; Webb & Graziano, 2015; Graziano, 2017), although the meaning there is perhaps narrower than what is proposed in this paper.

# Summary

todo

# Author Contributions

The author confirms being the sole contributor of this work and has approved it for publication.

# Funding

No funding was received.

# References

Beaudoin (1994)

1. https://en.wikipedia.org/wiki/Wernicke%27s\_area [↑](#footnote-ref-2)
2. https://en.wikipedia.org/wiki/Abstract\_syntax\_tree [↑](#footnote-ref-3)