Exploring Reinforcement Learning Models For Various Aspects of Decision Making

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in ***Electronics and Communication Engineering*** *by Research*

by

Gautham Venugopal
2018112003
`gautham.venugopal@research.iiit.ac.in`

International Institute of Information Technology
Hyderabad - 500 032, INDIA
May 2024

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled **“Exploring Reinforcement Learning Models For Various Aspects of Decision Making”** by Gautham Venugopal, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Adviser: Prof. Bapi Raju S

To - *Nature*

Acknowledgments

I would like to express my heartfelt gratitude to Prof. Bapi Raju for introducing me to the field of Cognitive Science and for the being an invaluable source of knowledge during my time at IIIT. I would like to thank Prof. Anuj Shukla and Krishn Bera for their mentorship during many of the projects I worked on at the Institute.

I look back fondly at the insightful conversations I have had with Prof. Don DCruz and Pramod Kaushik and I am thankful for their advice and guidance .I would like to thank my college mates Adarsh Dharmadevan, Joseph Cherukara and Arohi Shrivatsav for all their company and help they provided without which my time at IIIT would have significantly more worse off. I am thankful in having a friend in Belagavi Khadiravanna(BVK) with whom I could have hour-long conversations ranging from Machine Learning to Geopolitics. I am also thankful for the companionship provided by fellow CogSci Lab mates Rohan Reddy, Madhukar Dwivedi(Former) and Chirag Jain.

I believe the activities I have conducted as part of the Language Club and Ping will be among my most nostalgic memories of the Institute. I would like to thank Anvith Dosapathi, Sahil Bhatt and Aditya Shinde for all the help they provided. I would also like to thank Mahathi Vempati and Jaidev Shriram for their guidance and for their patience while working with me on various articles. I would also like to thank Radheshyam from Club Council for always being available for any help we needed, and Naren Akash for all the advice he provided to the Langauage Club team.

I am immensely grateful for the guidance and encouragement provided by Dr. Nicolas Rougier during my internship in France. I am also greatly thankful to Naomi Chaix-Eichel for all the help she provided with the coding and other technical aspects of the project. The experience gave me the confidence to step out of my comfort zone & also an opportunity to work with international colleagues.

I want to take a moment to thank my aunt Meera for her inspiration & encouragement while I was working remotely from home during the challenging quarantine period owing to Covid pandemic. Lastly, I want to express my deepest gratitude to my parents, whose support and belief in me have been a source of strength and motivation always.

Abstract

While designing computer applications to solve real world problems, engineers and Machine Learning researchers have the freedom to design models specific to the situation at hand. If the application is required to have high latency or low memory usage, the model is typically tweaked at the design stage itself to incorporate these properties, and such models cannot be used in cases where reliability or explainability takes precedence, even if the underlying task is the same. However, in the case of cognitive agents, the strategies and algorithms used to make decisions need to adapt to the situation in an online manner. A popular idea in cognitive science used to explain how animals make important tradeoffs is that of viewing Decision Making as an Evidence Accumulation process. By modelling Decision Making as sampling evidence until a threshold, it is possible for such models to showcase different behaviours as per the needs of the situation. In this thesis, we take the sequential sampling approach and combine it with Reinforcement Learning frameworks in an attempt to move towards a more comprehensive model of Decision Making.

In the first portion of the thesis, we explore how Linear Ballistic Accumulators can be incorporated as an action selection mechanism into Q-Learning models, which we refer to as RLLBAs. One important advantage this brings about is the ability of RLLBAs to utilise reaction time data in addition to choice data. We compare the performance of RLLBAs and conventional models in a non-trivial Grid Navigation Task with three action choices. It was found that RLLBAs were able to predict the actions taken by the subjects as well as conventional RL models while at the same time providing good predictions of the reaction time data. In addition, it was also shown that RLLBAs show significant differences in the goodness-of-fit between various forms of arbitration between Model-Free and Model-Based RL, something which is typically harder to achieve with choice data alone.

In the second portion of the thesis, we explore how evidence accumulation can be realized in RL neural networks. For this purpose, we take inspiration from existing literature on anytime neural networks and structured reservoirs. The central idea here is to structure the connections of the reservoir so that activity in the network propagates forward across time. As the activity propagates forward it undergoes more processing and becomes less noisy, meanwhile, as the output layer has access to earlier parts of the reservoir, the model can still respond quickly to sudden changes in the environment if relevant. On further experimenting with connectivity patterns found in the Basal Ganglia such as parallel pathways, we find that representing different inputs in different pathways based on the concept of stripes seen in working memory models offer superior accuracy in multi armed bandit tasks over conventional reservoirs.

Contents

List of Figures

Figure Page

List of Tables

Chapter 1

Introduction

1.1 Motivation

Life can be construed as a series of decisions. Often when we think of decisions, we often think of it simply as finding the right answer. However, Decision Making in the real world for biological agents are often far more complex. A lot of times with the information available one cannot deduce the correct answer but must resort to an educated guess. Effort put into processing information needs to be considered carefully so as to conserve energy. Finally, often, the time taken to come to final decision is also of paramount importance. In the domain of Cognitive Science, how animals deal with all these issues have been extensively studied and modelled. However a model that combines research from all these distinct areas has not yet been proposed. In my thesis, I make preliminary steps towards the goal of making a model that accounts for many different features of animal decision making. Almost as early as a century back, researchers had begun to identify that there are two different types of voluntary actions. Ones that are goal driven and ones that are habit driven. The primary difference between them being that goal driven actions are taken with a particular outcome in mind whereas habit driven actions are primarily responses to a stimulus which are performed without thinking about any particular aim. In the laboratory these two behaviours are often tested using the outcome devaluation task. Say that a mice has learnt to press a lever to receive a sugar pellet, chemicals are then added to the pellet in order to devalue the pellet. If the mice still presses the lever, that implies that the mice is undertaking in habit driven actions whereas if it stops doing so, then it implies that the mice was engaging in goal driven behaviour. Habit learning is important as it saves cognitive effort for frequently repeated actions. It provides working memory space so that agents can focus on other cognitively demanding processes while actions are taken automatically. However, there are many important questions to answer such as how are these two different processes for voluntary movements arbitrated. What if there is a conflict between habit driven and goal driven actions? While many models have been proposed for how this arbitration happens, it is hard to distinguish their fits to data when looking at choice data alone. In this work, we propose a novel model which can account for reaction time data as well. Another aspect of Decision Making is the Response Time. In the real world, there are many situations in which an action needs to be taken within certain time constraints. Thus the algorithms implemented in the brain needs to be flexible enough to adjust to various time constraints. Furthermore, in many scenarios, the reward rate becomes the crucial factor in survival. In such cases, it can make sense to make suboptimal decisions quickly than make optimal decisions at a slower pace. Finally, time and effort needs to be utilized depending on the difficulty of the problem at hand. Thus, in the case of making harder decisions, it would make sense to devote more time and effort than in the case of easier decisions. A popular set of models used to study how animals navigate these dilemmas are Evidence Accumulation Models. By incorporating the core features of EAMs into the models presented in this work, we propose models that are able to tradeoff between speed, accuracy and effort, while also taking into account underlying learning processes.

1.2 Contribution

A fundamental principle in both models that we introduce is Reinforcement Learning(RL) [83]. RL is a machine learning paradigm which focuses on agents learning from interaction with the environment through trial and error. As a theory whoose initial foundations were inspired from psychology [72], RL models have been popular in the cognitive psychology literature as normative models of reward based learning [58]. Such usage saw a particularly significant increase upon findings that found that phasic dopamine bursts in the midbrain encodes reward prediction error [78]. Today, RL based ideas have gone beyond decision making and learning to inspire ideas spanning multiple domains of cognitive science such as predictive coding [52]. In addition, it has also been found that neural networks trained using RL are more similar in activity to real brain networks [54].

RL models, however, do not predict response times directly. Response times have been correlated with the confidence with which people make decisions [33], and thus has significant potential to help us understand how people value different choices. Yet the inability of RL models to take these metrics into consideration means that researchers have to rely primarily on choice data alone while fitting RL models. This also restricts the ability of such models to incorporate non-skill based factors. For example, if a subject takes more time for motor execution but is otherwise able to perform the task well, this would be reflected in the response times but not in the accuracy. One solution would be to integrate RL models with Evidence Accumulation Models(EAMs) [28] that have been popular in Decision Making.

EAMs constitute a set of models that model Decision Making as accumulating evidence for picking one choice over the others until a certain threshold at which point the action corresponding to the choice is executed. By adding stochasticity to the accumulation process, randomness similar to that seen in human behaviour is seen in simulations of accuracy and response times. Initially, proposed by Ratcliff et al. [70] for modelling behaviour in memory retrieval task, since then such models have become mainstream in modelling perceptual decision making. They are supported by a wealth of research showing ramping brain activity in various brain regions during decision making particularly in the case of perceptual decision making [24] [81].

Popular models such as Drift Diffusion Model(DDMs) [70], are able to abstract more meaningful parameters from RT and choice data. For example, in the case of DDMs, rate of accumulation of evidence, termed drift rate, represents the quality of accumulation of evidence, which is indicative of the skill of sucject in performing the task whereas the threshold that is needed to be reached can be said to represent the response caution or response impulsiveness of the subjects. Both the above factors affect both the reaction speed and the accuracy of animal's performance in different ways and by using models like DDMs we can try to find the root differences in factors behind differential performances in subjects [69].

In this work we investigate how RL models integrated with EAMs are able to model behaviour in a sequential decision making task. RL and EAMs naturally complement each other by maintaining a clean dissociation between learning and decision making [53]. While RL concentrates on how neural representation in the brain are changed as a result of learning, DDMs focus on how these representations translate into actions. For evaulating these mdoels we used the Grid Sailing Task [29], where participants have to navigate from a start to end position using a non-trivial keymap consisting of three options. With regard to the EAM model, we used the Linear Ballistic Accumulator(LBA) [12] model a mathematically simplified version of DDMs which are more well-suited for multi-alternative decision making. We attempted various ways to link the two models and found that feeding the softmax of the Q-values as drift rates gave the best fits.

We found that RL-LBA models fitted to individual participants were able to predict their choices as well as conventional models, in addition to predicting response times with reasonable accuracy. We also found that due to using the additional reaction time data, RL-LBAs were able to find significant differences in goodness-of-fit between various models of arbitration between model-based and model-free RL. As different models of arbitration did not often imply significantly different predictions in choice, it is difficult to get significantly different predictions with choice data alone.

However, the above model is an abstract psychological model. While they explain the algorithms used by animals, exploring how they are implemented in the brain would help us better understand Basal Ganglia(BG) based psychological disorders such as Parkinson's. Thus, in the second part of this work, we explore using Echo State Networks(ESN) inspired from BG connectivity structures.

ESNs [40] are neural networks based on the paradigm of reservoir computing. The central idea here is to project the input into a reservoir layer with a significantly larger number of neurons than the input dimension. The reservoir has connections between neurons in the layer, thereby making the network recurrent. By setting both the input connections and reservoir connections randomly, the idea is to create high dimensional representation of the input, where every neuron in the reservoir is a distinct function of the input. On top of this, an output layer is trained to choose the relevant neurons necessary to perform the task. The ESN offers several advantages over conventional neural networks such as lack of back propagation over multiple layers and a closed form solution for determining the output layer connections. Their biological plausibility makes them attractive for modelling in cognitive neuroscience.

The BG [71] is one among the many sub cortical structures in the brain which have maintained distinct local connectivity patterns through many levels of evolution. The BG consists of parallel loops interacting across multiple regions of the brain to decide which action to take. In recent times, structured reservoirs [25] with either distinct reservoirs connected together in specific manners or a single reservoir whose connections are structured in a particular manner have become popular in the machine learning literature and the cognitive neuroscience literature. We tested the performance of such structured reservoirs inspired from BG on a temporally challenging multi-armed bandit task where models had to remember inputs presented earlier and also had to respond quickly to new inputs. We found that architectures based on O'reilly's model of working memory [60] using stripes showed superior performance to other connectivity patterns including a standard reservoir.

1.3 Thesis Overview

The thesis consists of 6 chapters including the Introduction and Conclusion. Chapter 2 deals with Reinforcement Learning, and covers its fundamental principles and contributions to Neuroscience and Machine Learning. Chapter 3 looks at the current popular approaches of looking at Decision Making time and covers the mathematics behind Evidence Accumulation Models and introduces Anytime Neural Networks. The RL-LBA model is explained in detail in Chapter 4 and the results of various arbitration methods are compared with conventional models.

Chapter 2

Reinforcement Learning Models in Cognitive Science

Humans and other animals learn many fundamental skills not from any teacher but by interacting with the environment. They choose to go to different places, and follow different strategies [77]. Then, based on the outcomes that result from those actions, they make various associations, form many heuristics [16] and assign values to different objects and places. How animals are able to perform such feats efficiently in uncertain situations and noisy environments is a popular subject of research.

Since we understand that animals evolved as subject to natural selection, which propagated further along the genealogy, those features which contributed to fitness and survival, we assume that animals, especially humans, must perform tasks of ecological validity with near optimality. Thus, a popular approach to studying human behavior is from a normative account, where we look at what would be the most optimal way to solve a task or a problem, and then, try to work backwards to figure out how the brain adapted to implement the same or similar computations. In the case of trial and error learning as described above, the typical normative model used is that of Reinforcement Learning(RL). In this chapter, we will look at some of the basic ideas in RL, type of RL and how it relates to the brain.

2.1 Reinforcement Learning: Central Ideas

2.1.1 From the Computational Perspective

Reinforcement Learning(RL) [83] is a Machine Learning(ML) paradigm, where an agent aims to maximize the cumulative rewards it gains from the environment. Typically, the agent is given information about the current state and upon choosing an action to perform, is shown the consequences of that action in terms of change of state and a reward recieved. The above sequence may be referred to as a trial or a step, and combination of several consecutive steps is called an episode, across which RL algorithms hopes to maximize cumulative rewards.

RL brings about special challenges that needs to be answered by prospective algorithms, which are not present in other ML paradigms such as supervised or unsupervised learning. First of all, unlike in supervised learning, in RL, a teaching signal telling the model the correct option is not present. One solution to this problem is to simply explore every possible action that can be taken at every possible state, i.e, explore every possible episode. However, in situations where there a limited number of episodes available for training and there is not enough opportunity to exhaustively search for possibilities, this would not be the optimal strategies. In life too one often does not have the opportunity to consider every possible strategy, but at the same time not exploring options at all might end up causing agents to follow sub optimal policies. Thus good RL algorithms need to balance between Exploring options and Exploiting rewards, a feature commonly termed the Exploration-Exploitation Tradeoff.

Credit Assignment is another challenge faced by RL algorithms. It is similar to the challenges faced by supervised learning algorithms in that the models have to be able to recognise which features of the input are relevant for making the correct prediction. In the case of Natural Language Processing models, for example in the case of an emotion recognition task where sentences have to be predicted as conveying one or the other emotion, the model would have to look at all the input spread across temporally, identify relevant phrases or clauses and make appropriate predictions. RL algorithms are often forced to solve a slightly more complicated version of the above problem in that they not only have to look at what has passed to decide the right action to take, but also look at the future. Credit Assignment in RL is not only about deciding what action to take at the current state, but also about deciding what actions would lead to better states in the long term future.

2.1.1.1 Model Formalisms

The problem statements that RL algorithms are required to solve are typically defined in the form of Markov Decision Processes(MDPs). MDPs are a set of computational elements that collectively characterise the environment the agent interacts with. These elements consists of:

1. **State Space(S):** Set of all possible states that can be reached by the agent. For the rest of this work, the current state shall be referred to as s and the next state as s'.

2. **Action Space(A):** Set of all permitted actions that the agent can take. For the rest of this work, the action undertaken at the current state s, resulting in transitioning to state s', shall be referred to as a.

3. **State Transition Probabilities(P(s,a,s')):** Defines the probability that taking action a at state s will result in a transition to state s'.

4. **Reward Function(R(s,a)):** Defines the reward that will be obtained by the agent upon performing action a while in state s. We shall refer to the reward obtained upon performing an action during the current state as r. For a given episode, we shall refer to the reward obtained after the first step as r_1, second time as r_2 and so on.

For example in the case of chess, the state space S, would consist of all possible configurations of the pieces, except the ones that can not be meaningfully be reached in a fair game. For example configurations where there are two bishops of the same colour on a diagonal or a pawn on the starting square

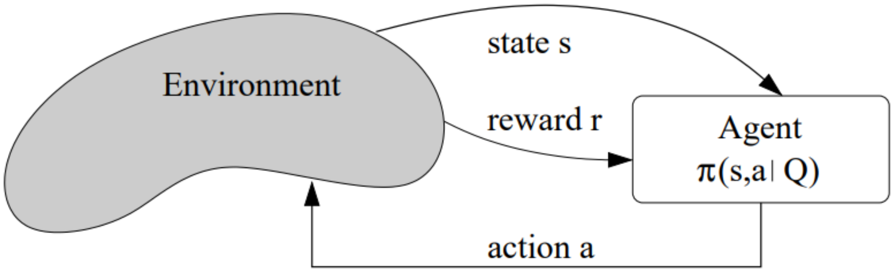


Figure 2.1 In the RL paradigm, agents interact with the environment via actions. After each action, the agent arrives in a different state and is given a reward depending on the action taken. Figure taken from [82].

would not be valid configurations and thus, would not be part of S. The action space A would consist of all possible legal actions in the chess game. The Reward function R(s,a) can denote a positive reward upon checkmating your opponent or a negative reward upon getting checkmated by your opponent. The reward function essentially dictates the objective of the task, however the reward function and the objective does not necessarily always have a one-to-one relationship. In the above case, we use a sparse reward representation, however such representations can often be hard for RL algorithms to work with, so we often tune it to drive the learning in a particular way. For example, we may give a negative reward for each piece lost. This would incentives the agent to not too loose any pieces, which would make it easier for the algorithm to learn how to play chess as it is a winning strategy to not loose too many pieces. At the same time, this might also cause the algorithm to play more defensively without making any sacrificial positional plays.

The main objective of RL algorithm is to maximize the cumulative reward gained. Referring to this quantity as G, we can define it mathematically for episodes with non-limited number of trials as:

$$G = r_1 + r_2 + r_3 + ...$$

To do this we essentially need to come up with strategy for how to act in different situations. This is formally referred to as the policy followed by the agent. The policy, $\pi : S- > A$ is a mapping from the state space to the action space, essentially dictating what action to choose at a particular state. Policies too like state transition matrixes can be deterministic or stochastic. If they are stochastic each state would map to a probability distribution(random variable) over the action space.

However, depending on the nature of the reward function, it is possible that this quantity can go to infinity. Thus we modify the equation with a discount factor, γ, as follows:

$$G = r_1 + \gamma r_2 + \gamma^2 r_3 + ...$$

Most modern algorithms work on the basis of assigning values to states, with higher value states being more desirable to be in. In such cases, value of a state is typically defined as the amount of discounted reward that can be obtained on performing optimally from that state. Thus, we define value of a state s, V(s) as :

$$V(s_t) = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + ...$$

,where t represents the current time step in the episode, s_t and r_t represents the current state and the reward obtained at the current step, and r_{t+1}, r_{t+2}, represent rewards obtained from subsequent actions by following the policy.

An early idea that was popular in RL literature was that once we could get an estimate of the value of all the different states, the optimal policy would always be to just choose the action that would lead to the best possible state. The value could be estimated by methods such as Monte-Carlo Tree Search, but more popularly it has been shown that the value can be updated using update rules such as below:

$$V(s_t)_{old} = V(s_t)_{new} + \eta \cdot \delta_t,$$

where δ_t represents the state value prediction error experienced at time t, given formally as:

$$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$$

The central idea here is that $r_t + \gamma V(s_{t+1})$ represents the discounted reward that the agent can expect to get now that it has got reward r_t and transitioned to state s_{t+1}. As it can be proven that repeatedly using the above update rule while traversing across the state space would lead to the values converging to the ground state values, this strategy forms a fundamental basis for many popular RL algorithms including state-of-the-art models.

2.1.1.2 Q-Learning

While all this sounds good, an important question that arises now is what about action selection? If we have information about the state transition matrix with access to some kind of a world model, we can simply choose the action which would lead to the next best state. However, in ecologically relevant circumstances, we find that many animals are able to perform reward based learning even without a world model. The idea behind implementing this computationally is to compute a value at the action level for each state. We call this term Q-value, represented by:

$$Q(s_t, a_t)_{old} = Q(s_t, a_t)_{new} + \eta\delta_t$$

, where $Q(s_t, a_t)$ represents the value obtained from choosing action a_t at state s_t and δ_t once again represents the value prediction error given by:

$$\delta_t = r_t + \max_a \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)$$

Here the max operator means that the prediction error is computed with respect to what is believed to be the best action one could take from state s_{t+1}. This method is referred to as off-policy updates as the optimal action, in this case, a_{t+1} might not be the action that is actually taken in the next state. Alternatively, we can also define Q-values as:

$$\delta_t = r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)$$

Here $Q(s_{t+1}, a_{t+1})$ represents the next state that was actually encountered by the agent. Of course, this means that the update rule cannot be applied in a truly online sense, but only after it has encountered state s_{t+1}. This type of Q-value updating is called on policy updates.

Q-Learning [83] is an RL algorithm where we traverse through the environment while constantly updating the Q-values of each state action pair encountered. Essentially we start with a Q-value table initialized to zero which gets updated as the agent explores the environment. As we are working directly with the Q-values, the optimal policy would simply selecting the action with the highest Q-value, however we still have to deal with the Exploration-Exploitation problem. A popular way to face this challenge is to use a method called Epsilon-Greedy action selection. The central idea here is to randomly choose an action(explore) a set proportion of times and select the best action(exploit) other times. In the above method we have a variable ϵ, which designates the probability with which the agent chooses a random action at each step. While ϵ can be kept constant throughout, it can also be varied across the session, with earlier episodes having epsilon and thus high exploring as comparison to later episodes being more about exploitation.

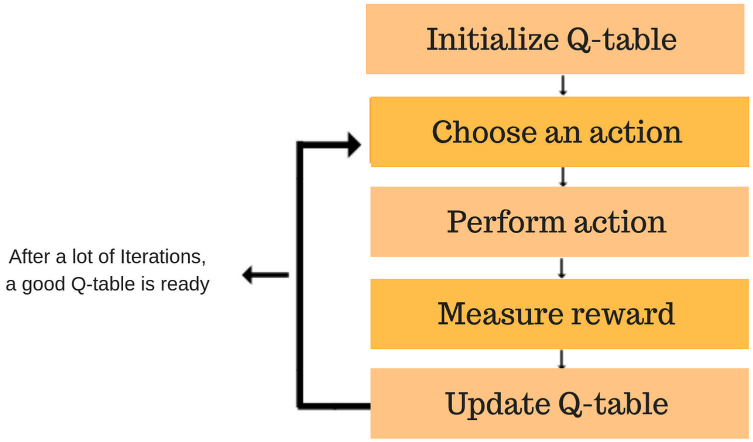


Figure 2.2 An flow-chart of the various steps in Q-Learning

Many state-of-the-art learning algorithms such as Deep Q-Learning are based on Q-Learning. Such algorithms have been able to beat human performance in many games such as Atari and also forms an important part of training many chatbots. Important computational aspects of Q-Learning have also signs in neural correlates in animal brains as we will see in the upcoming session.

Algorithm 1 Q-learning (off-policy TD control) for estimating $\pi \approx \pi_*$

```
Algorithm parameters: step size $\alpha \in (0, 1]$, small $epsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(\text{terminal}, \cdot) = 0$
for each episode do
    Initialize S;
    for each step of episode do
        Choose $A$ from $S$ using policy derived from $Q$ (e.g., -greedy)
        Take action $A$, observe $R$, $S'$
        $Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma \max_a Q(S', a) - Q(S, A)]$
        $S \leftarrow S'$
    end for
end for
```

2.1.2 Neuroscience

The Pavlovian experimental paradigm [62] was a major focus of experiments on learning in the 70s. A typical experiment of the sort, would have a two or more stimuli one of which would temporally precede the other. One of the stimuli, would evoke a response from the animal. For example in the iconic original Pavlov experiment done with dogs, this stimuli was food and the response it evoked was salivation. This stimuli is typically called unconditioned stimuli(US) and the response is referred to as unconditioned response(UCR). The other stimuli would be a neutral stimuli(NS) such as the ringing of a bell which would typically not elicit any significant response from the animal.

By ringing the bill before providing the food, over time the conditioned response(CR) starts being shown by the dog in response to the previously neutral stimuli. The CR reflects the animal's expectation with regards to encountering the US. Essentially, a predictive relationship is learned by the animal associating the NS with the US. Once this relationship is learned, we refer to the neutral stimulus as the conditioned stimulus.

Initially, there was a significant amount of literature which associated dopamine with being a 'reward signal' of sorts [100]. However neurophysiological studies on the basis of Pavlovian experiments disproved this theory. Experiments showed that there was an increase in phasic dopaminergic activity upon encountering the US or reward initially, however, the activity decreased over time until it ceased to exist over time [78]. Instead, over time phasic dopaminergic activity was observed when the animal was exposed to the NS or cue instead. Soon, connections were made with computational neuroscience, and it became established that rather than representing reward, dopamine really represents reward prediction error, the difference between expected pleasure and experienced pleasure. Overtime, further nuanced experiments were conducted such as removing the US after the CS was learned. As expected, there was an increase in dopamine activity in the midbrain reflecting negative reward prediction error.

Although an in-depth discussion of learning systems in general is out of the scope of this work, it would helpful for prividing useful context for the next chapter. The popular theory today is that different parts of the brain are specialized for different type of training. Namely, that the Basal Ganglia(BG) is specialized for reward based learning, cerebellum for supervised learning, and the cerebral cortex for unsupervised learning. This is based on the local connectivity patterns and neurological signals found represented in these regions across species. In the case of BG this includes the multiple parallel pathways that pass through the thalamus and the cortex. It is believed that when making decisions each option is compared on the basis of morphologically distinct Go-No Go pathways in the BG, whoose connection weights are modulated by dopamine neurons in the substantia nigra.

2.2 Model Based Reinforcement Learning

As we saw earlier, Q-Learning does not use any world model and does not depend on any predictions of state transitions. However, similar to how there are cases where animals must perform reward based learning without knowledge about the environment, there are also cases where animals use existing knowledge to plan ahead. The type of RL where the agent has access to state transition probabilities and reward function and can make use of it to plan ahead is known as Model Based Reinforcement Learning(MB-RL).

In the Machine Learning side, MB-RL is not often implemented as an online algorithm. If one knows about the dynamics of the environment, they can simply simulate possible strategies in their mind to estimate the optimal policy. Methods that do this such as Value Iteration and Policy Iteration are well studied. However, on the cognitive psychology side, we use concepts in graph traversals to try to understand how animals use environmental knowledge to plan ahead.

One popular such algorithm is Breadth First Search(BFS). BFS is a graph traversal algorithm where starting from a parent node, we first traverse to each node connected to said node, and then repeat the same process for the set of nodes that was traversed. The set of nodes that were traversed in the first step are said to be at a depth of 1. Similarly, the new nodes that we come into contact while applying the same process for the prior set of nodes are reffered to as depth-2 nodes. The central idea here is that instead of comparing the Q-values of the parent node, we look at the state values of all nodes within a certain depth, and then select the action that will lead us to the node with the best value.

For example, let us say that when playing chess you would like to understand which way to move one of your horses. One way to look at it, would be think about all possible positions that can be reached by moving the horse one step and then assessing their value individually. This would correspond to depth-1. Alternatively you can look at all possible positions that can be reached by the horse in two steps, and take the values of each of those states into consideration while making your move. In this way, we use information about how the horse moves and information about the values of various positions that may be encountered in the future to take a decision in the current time step.

2.3 Hybrid Models

2.3.1 Computational Perspective

We have looked at both MF-RL and MB-RL, and understood the primary difference between them: the use of a world model. However, difference don't stop there. As a consequence of using more information, MB-RL is typically able to learn better policies more quickly than MF-RL, especially in complex environments. However, when it comes to action selection, MB-RL takes more computation and time than MF-RL per time step. Often, there may be situations where it may be useful to choose one over the other and studies have shown that depending on the task at hand, there can be effective ways to combine both methods to improve performance. In this section, we look at methods of arbitrating between MF-RL and MB-RL that have been suggested in the literature.

One relatively straightforward method of arbitrating between MF-RL and MB-RL that became popular initially is weight based arbitration [36]. The idea is to maintain two Q-value tables. One that is always updated using a simple Model Free update, whereas the other is always updated using a Model

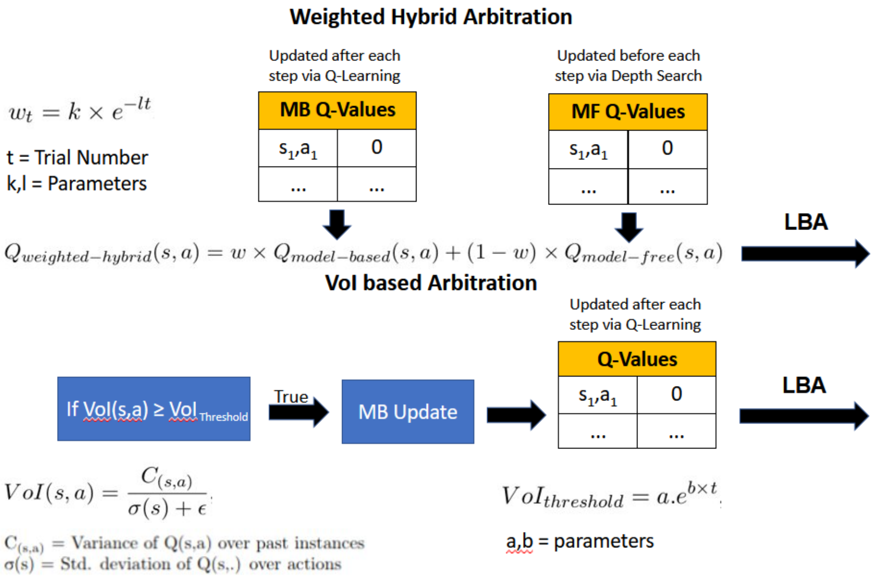

Weighted Hybrid Arbitration
Updated after each step via Q-Learning
Updated before each step via Depth Search
$w_t = k \times e^{-lt}$
t = Trial Number
k,l = Parameters
MB Q-Values
MF Q-Values
s1,a1
0
...
$Q_{weighted-hybrid}(s,a) = w \times Q_{model-based}(s,a) + (1-w) \times Q_{model-free}(s,a)$
LBA
VoI based Arbitration
Updated after each step via Q-Learning
Q-Values
If VoI(s,a) ≥ VoI Threshold
True
MB Update
LBA
$VoI(s,a) = \frac{C_{(s,a)}}{\sigma(s)+\epsilon}$
$VoI_{threshold} = a.e^{b\times t}$
C(s,a) = Variance of Q(s,a) over past instances
σ(s) = Std. deviation of Q(s,.) over actions
a,b = parameters

Figure 2.3 Illustration of the working of the Arbitration Mechanisms. In the case of Weighted Arbitration, two Q-value tables, one updated by a Model-Based algorithm and the other updated by a Model-Free algorithm is maintained. However, in the case of Value of Information based arbitration, only one Q-value table is maintained.

Based update of a certain depth. During action selection, a weighted linear combination of both Q-values are taken as the final Q-values used for deciding which action to take. The weight may change across the episodes. For example, weights used in earlier episodes may prefer MB-RL and the weights used in later episodes may be tilted towards MF-RL. It is important to note that using MB-RL earlier can offer significant performance increases as the model can start exploiting more easily. Earlier experience performing optimally with MB-RL helps the MF-RL Q-values converge more quickly.

While weight based arbitration can be a good abstract model of arbitration processes in the brain, it suffers from a serious defect. At least in the computational model, the trade-off of pros and cons between MB-RL and MF-RL is not being used optimally, as both updates are used at every trial. Ideally, one could argue, MB-RL updates should primarily be used when such updates actually make a difference in policy. This is the central idea in Value-of-Information(VoI) [7] based arbitration.

In this kind of arbitration, we look at the value of the information that can be expected to be gained by using by MB-RL i.e, by simulating future steps, and then if it is above a particular threshold, we use an MB update. Essentially, if you are confident that you have accumulated sufficient experience to trust your gut instinct, you would not spend much time looking ahead. Thus in this method, we have only one Q-value table. By default, the table is updated each time a reward is received using a model free update. However, at the time of action selection, the value of information expected from simulating ahead is estimated and compared to a threshold. If the value is higher, then the agent simulates possibilities until a particular depth and accordingly updates the values of the Q-Value table.

When compared to Weight Based Arbitration, VoI based arbitration has the advantage that it is more computationally efficient and actually makes more sense from a trade-off perspective as we would be saving time in those trials in which we choose not to simulate ahead. It is important to note that when talking about interaction between MB-RL systems and MF-RL systems, it is not just about arbitration, but there are ways in which both systems can cooperate to achieve different goals than just performance.

2.3.2 Neuroscience

The idea of the brain having two different systems of learning is very popular in the cognitive science literature over various domains [58]. The most generalized idea is that for many sorts of brain processes, there is a system that learn slowly but once learned can execute action quickly and a system that requires time and attention during decision making but dosen't require much experience or time for learning and is significantly more adaptible. This generalized view has found some proponents in various domains including metacognition and hippocampus memory replay.

However, in this work, we will primarily be looking at how this idea has been used for modelling motor skill learning [18]. In this case, the dichotomy was able to transpose itself neatly over the goal-directed vs stimulus driven decisions making behavioral paradigm that had accumulated much evidence.To put simply, goal directed actions are those actions taken expecting a particular outcome, whereas in the case of stimulus driven actions are primarily just responding to environmental cues without expecting anything ahead. While stimulus driven actions are learned through repeated experiences and develop as habits, goal directed decisions are made using previously learned information and using heuristics and can use semantic knowledge to great effect.

2.4 Neural Network Models - Deep Q-Learning

A central element to traditional RL algorithms is the Q-value table, whoose size is determined by the number of actions and number of states. To a certain extent, even when state spaces and action spaces are continous, cleverly quantizing the spaces into bins have been shown to be effective solutions. However, as state spaces and action spaces become increasingly larger, not only does the Q-value tables take up a lot of space, but by treating the values for each state-action pair as completely different from one another, we miss opportunities for generalization across states.

In fact, in the initially popular Rescorla-Wagner model, the focus was on features of experience rather than temporally separated states. The advantages offered by such a perspective can be intuitively understood. For example, when making an algorithm to assess the value of various chess positions, it would be inefficient to consider each state separately. A better way would be to consider features such as the presence of Rooks on the seventh rank, or an outpost on the sixth rank. In fact modern state-of-the-art chess playing agents, such as Stockfish use similar heuristics to understand the value of various positions.

One way of moving towards a feature space while keeping the fundamental concepts of Q-Learning is to use neural networks as a function approximators for the Q-Value table. Essentially the idea is to use a neural network for the mapping from state-action space to value. This allows for feature space based generalisation. For example, relevant objects in an image can be encoded in activations of intermediate neurons which can then become factors in deciding the values of different states. This is the central idea behind Deep Q-Learning [55] networks and the reservoir networks we will analyse in chapter 4.

Chapter 3

Modelling Response Time in Decision Making

When approaching cognitive science from a Machine Learning background, it can be easy to not give as much importance to reaction time and concentrate primarily on optimality without sacrificing accuracy. While in computer science, researchers focus on how to make accurate provable algorithms that are more computationally efficient, research on human behaviour has gone in a different direction in viewing accuracy, energy used and time taken as a trade-off. This approach to thinking about cognition is commonly known as bounded rationality.

There are intuitive ecological situations where we can expect response time to play a major role, particularly in when there are time constraints. In situations where there is a limited amount of time to make a decision(say for example, in the case of an encounter with a predator), animals must be able to choose a near-optimal if not at least reasonable action quickly. However, the issue is not just about time constraints. Even if you have an infinite amount of time, it can still be beneficial to use suboptimal policies if you are able to choose actions more quickly. Even when compared to a policy that objectively is more rewarding, using a more quickly implementable suboptimal strategy can result in higher reward rate. In addition to constraints, with respect to time, there are constraints with respect to physiology too. Despite the large number of neurons present in the brain(86 billion). the brain is still constrained by limits of metabolism and energy uptake. All of this forces the brain to not only need to have mechanisms to effectively utilise tradeoffs such as speed vs accuracy, but for the mechanisms of storage and retrieval be flexible enough to support such tradeoffs.

Despite the ideas of bounded rationality [59] being explored in the literature for quite some time, much of the discussion has revolved around understanding how it affects heuristics and other explicit strategies. There is much potential in trying to understand how these ideas would constrain neural systems and architecture especially since this kind of analysis is not popular on the Machine Learning side. In this section, we will look at abstract psychological models that take into consideration and make proposals as to how they can be translated into neural systems

3.1 Evidence Accumulation Models

Let us suppose that you are to design a sensor for a car. The sensor collects information from the front of the car, and ideally, if the car is about to hit something, the sensor needs to send a signal and activate emergency breaks. As the sensor information is noisy, there is a tradeoff to be made. We don't want false negatives as it would be bad for the car to clash with something, whereas we don't false positives as it would make for a bad driving experience.

A classical way of solving this in Signal Detection and Estimation Theory is by using something called Wald Sequential Probability Ratio Test(SPRT) [95]. The SPRT takes all relevant sensor data in samples, then assesses how each sample supports or rejects the possibility of there being something in front of the car. When implemented in an online manner with a queue, it works similiar in essence to a Kalman filter, accumulating evidence until a certain threshold at which a signal is said to be detected. In this case, when the threshold is reached the emergency braking system is triggered. Higher threshold would mean less false positives, wheras a lower threshold would mean less false negatives.

Research in cognitive science have found results suggesting that animals too use similar evidence accumulation mechanisms for decision making. [28] In an well-cited experiment, Shadlen and Gold [37] found ramping activtiy in the Frontal Eye Field region of the primates brain when performing a random dot motion coherence task. The rate of ramping of the activity was proportional to the reaction time of the primates. This suggests that when making perceptual decisions we sample information from the environment to accumulate evidence until we are confident enough to make a decision. As our skill in a task increases, we are able process more information more quickly more confidently which would be indicated in the rate of ramping of activity. The threshold here would indicate response caution.

3.1.1 The Mathematics - Wald Sequential Probability Ratio Test

Let us go back to the example discussed earlier. Let us say that you want to stop the car if you are at least 95 percent sure that there is an obstacle in front of the car. This would mean that your desired error rate $\alpha = 0.05$. We will now look at the mathematical properties of this situation.

Let O be the event that there is an object in front of the car and NO be the event that there is not. Let $s_1, s_2, s_3, ...$ be the sensor samples received at relevant time steps. Then we can say that ideally,

$$\frac{P(O)}{P(NO)} > \frac{1-\alpha}{\alpha} \tag{3.1}$$

Upon observing a sample s_1, we can rewrite the above equation as:

$$\frac{P(O|s_1)}{P(NO|s_1)} > \frac{1-\alpha}{\alpha} \tag{3.2}$$

To calculate the left hand side, we can use the Bayes rule which is defined as follows:

$$p(a|b) = \frac{p(b|a)p(a)}{p(b)} \tag{3.3}$$

This gives us,

$$\frac{\frac{p(s_1|O)p(O)))}{p(s_1)}}{\frac{p(s_1|NO)p(NO)}{p(s_1)}} > \frac{1-\alpha}{\alpha} \tag{3.4}$$

By cancelling $p(s_1)$ from denominator and numerator, we get,

$$\frac{p(s_1|O)p(O))}{p(s_1|NO)p(NO)} > \frac{1-\alpha}{\alpha} \tag{3.5}$$

Now, upon observing another sample s_2, which we will assume to be independent of the first, we get the followung upon application of Bayes rule:

$$\frac{p(s_2|))p(s_1|O)p(O))}{p(s_2|NO)p(s_1|NO)p(NO)} > \frac{1-\alpha}{\alpha} \tag{3.6}$$

As this can be quite complex to work with, we take the logarithm of both sides, giving us:

$$\log(\frac{P(O)}{P(NO)}) + \log(\frac{P(s_1|O)}{P(s_1|NO)}) + \log(\frac{P(s_2|O)}{P(s_2|NO)}) > \log(\frac{1-\alpha}{\alpha}) \tag{3.7}$$

Thus, in general we can express the policy for choosing 'O' as calculating the sum of the logarithm of the ratio of prior probabilities plus the sum of the log-likelihood ratio of each of the independent samples of evidence observed so far and comparing that to a criterion that depends on our desired level of accuracy. Therefore, we can define a "decision variable" x_O for choosing O as:

$$x_O(n) = \log(\frac{P(O)}{P(NO)}) + \sum_{k=1}^{n} \log(\frac{P(s_k|O)}{s_k|NO}) \tag{3.8}$$

We do the same for choice 'NO' and then update these variables using samples of evidence until one of them crosses the criterion $C = \log(\frac{1-\alpha}{\alpha})$. This is known as the "sequential probability ratio test" (Wald, 1945), and it minimizes the number of samples needed to reach the desired level of accuracy. The sequential sampling models popular today in decision making literature are inspired by this equation.

3.2 Congitive Models

The most common metrics of performance in any task is accuracy and reaction time. However, at the end of the day, these are derived variables and often not directly the factors that we are interested in. By having a model of decision making that can be fitted to data, we are able to get more meaningful insights from behavioural data.

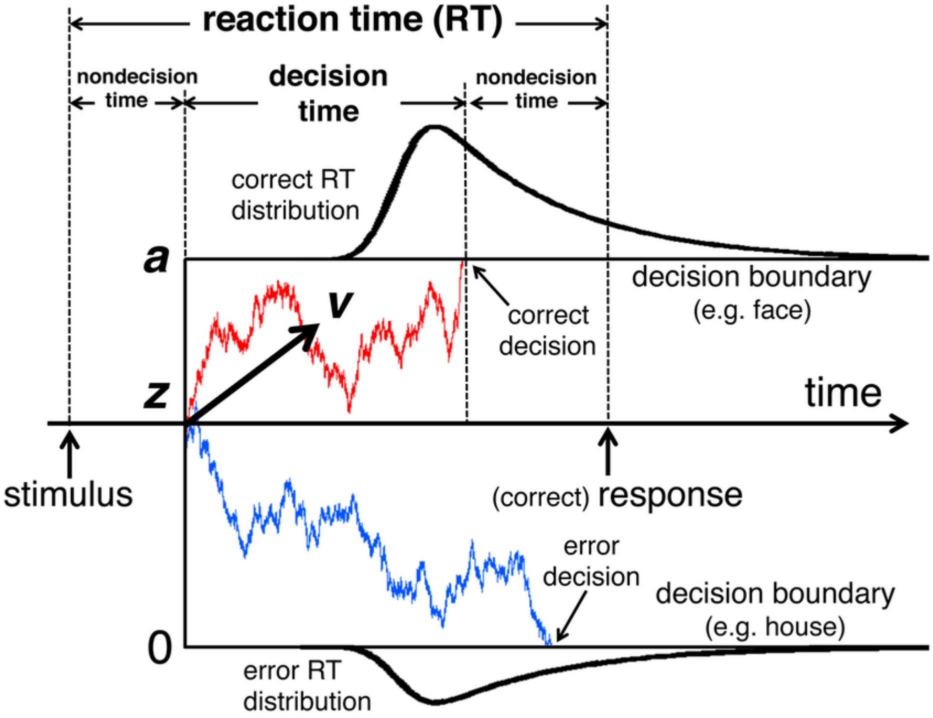


Figure 3.1 An illustration of the working of the Drift Diffusion Model.

3.2.1 Drift Diffusion Model

In the Drift Diffusion Model [68] [66], decision making is viewed as a process of accumulating evidence until a threshold is reached. For 2-choice tasks, this process is modelled by the evolution of a decision variable, where a decision is taken when the variable reaches a threshold. Typically the two choices are represented by a positive and negative threshold and the threshold reached by the variable indicates the choice taken. The time steps that it takes for the variable to reach the thresholds is added to a non-decision time to calculate the time taken for deliberation. The starting point of the decision variable indicates a bias for either option.

The equation that determines the evolution of the decision variable, x, is as below:

$$\tau \dot{x} = v + \varepsilon(t), \tag{3.9}$$

where τ represent the time constant, $\dot{x}$ represents the change in the decision variable, u represents the drift rate, and $\varepsilon(t)$ represents random noise.

Each parameter in the DDM has implications for cognition. The drift rate, u, is analogous to the summation term in Equation 3.8, and is a measure of one's skill in the task. A skilled person would be able to accumulate information more quickly and would need less thinking to settle upon a decision. The threshold represents the risk taking tendency or the cautiousness of the suject. The non-decicison time represents the time taken for perception and motor execution and is typically fitted for each participant and kept constant thought the session. Finally, the variation in the starting point aims to model the bias for either of the two options. This bias can either represent a trial-independent preference, such as liking for a particular shape or color, or an attentional effect such as in the posner task.

Models like DDMs allow us to go beyond RT and accuracy to measure important undelying factors. Let us take the following example. Consider two people Jack and James answering a given test. Let us say that Jack got 90 percent of the questions correct in 2 hours while James was able to get 70 percent of the questions correct in 1 hour. How can we know who is better at solving the questions? A clear answer cannot be gained from considering accuracy or time taken alone. In this case by modelling their behaviour and comparing the fitted values of their drift rates, we can come to a reliable estimate of their skills in answering the tests.

DDMs also draw attention to the fact that skill is not the only factor that determines accuracy in a task. Radcliff et al. [69] tested the performance of two groups, young and old in a verbal proficiency task. It was found that older people performed more slower than the younger group. However it was found that this is not due to performance but due to them being more risk averse and other factors. It was found that when DDM models were fit to the participants of both the groups, there was no significant difference in drift rates. However, the older group had significantly higher thresholds and non-decision times.

DDMs have been very popular for modelling decision making in perceptual and memory retrieval tasks and are finding increasing acceptance in modelling value based decision making. This is primarily due to the goodness of fit they provide. In fact not many mathematical models provide the distinct right skewed distribution of RTs seen in human behaviour for a wide range of parameters. However they also come with certain disadvantages, namely they can sometime be more complex to work with especially when if one needs to modulate any of the parameters with respect to a task parameter. IN addition they also cannot be easily expanded to multi alternative decisions, even though various proposals have been suggested. In fact many alternative EAM models popular in the literature are DDM models minorly modified to solve certain problems [91].

3.2.2 Linear Ballistic Accumulators

Linear Ballistic Accumulators(LBAs) [38] provide a solution for many of the problems faced with DDMs. LBAs are a mathematically simplified version of DDMs. While in DDMs, the random noise is added at each time step in the case of LBAs, the stochasticity directly affects the parameters. this means that while simulating the model, one does not need to run it over multiple steps, but can get the results in a one-shot manner.

Unlike DDMs, where the decision variable represents competition between two choices, in the case of LBAs, each choice has a separate accumulator that represents it. When one of the accumulator reaches a threshold the action corresponding to said model is said to be taken and the intercept of the drift with the threshold is said to be the reaction time.

There are two sources of stochasticity in LBAs. First, the starting point is drawn from a uniform probability distribution:[0,A] where A is a parameter. Secondly the drift rate is drawn from a Gaussian probability distribution:(u, σ), where u represent the mean drift rate and σ represents the standard deviation of the Gaussian distribution. Interstingly, we find that the starting point uniform distribution interacts with the drift rate Gaussian distribution to form the charecteristic right skewed distribution of RT when simulated. It also should be noted that the LBAs can be mathematically proven to be simpler models of DDM that retain much of their properties.

3.3 Anytime Neural Networks

While abstract models of decision making which incorporates RTs such as DDMs and LBAs have been well studied, ideally we hope that studies with these models would provide insights into the working of the brain and help us build neural network models. One popular problem in the field of Machine Learning which has the possibility helping with this is anytime neural networks.

Anytime algorithms [99] have been a subject of study since the start of artificial intelligence. They are decision making or classification algorithms that can be stopped in between to get a reasonable estimate than having to wait for a fixed amount of time. Interestingly, the advantages that anytime algorithms provide are quite relevant to ecological settings.

For one, anytime algorithms provide a solution to the time constraint problem as if a need arises to make a decision immediately, such algorithms can provide a reasonable estimate. Anytime algorithms also naturally adapt to the difficulty of the task. In the face of an easy problem, anytime algorithms can give answers quickly without much computation, whereas for harder problems they can appropriately give more compute. For this reason, these kind of algorithms are increasingly being used in search engines, where a limited compute has to be distributed among all the searches. However, beyond applications in ML, we find that these advantages are very relevant to animals too, who take more time depending on the difficulty of the decision to be made.

Recently, there has been renewed interest in exploring neural network models of anytime algorithms. Of these, there are two types: Recurrent Networks which process the input repeatedly, and Feedforward Networks with early stopping mechanisms. In the former, the same input is repeatedly processed by a recurrent neural network and the final hidden state used for calculating the output is a linear combination of all previous hidden states. The weights are calculated by a separate neural network and the process stops when the weights sum upto one. In the latter, similiar in architecture to the Inception model, early stopping layers are added to the intermediate layers of a neural networks. These early stopping layers try to predict the label by just using the intermediate layer representations. If these layers are sufficiently confident in their predictions, then no further computation is performed.

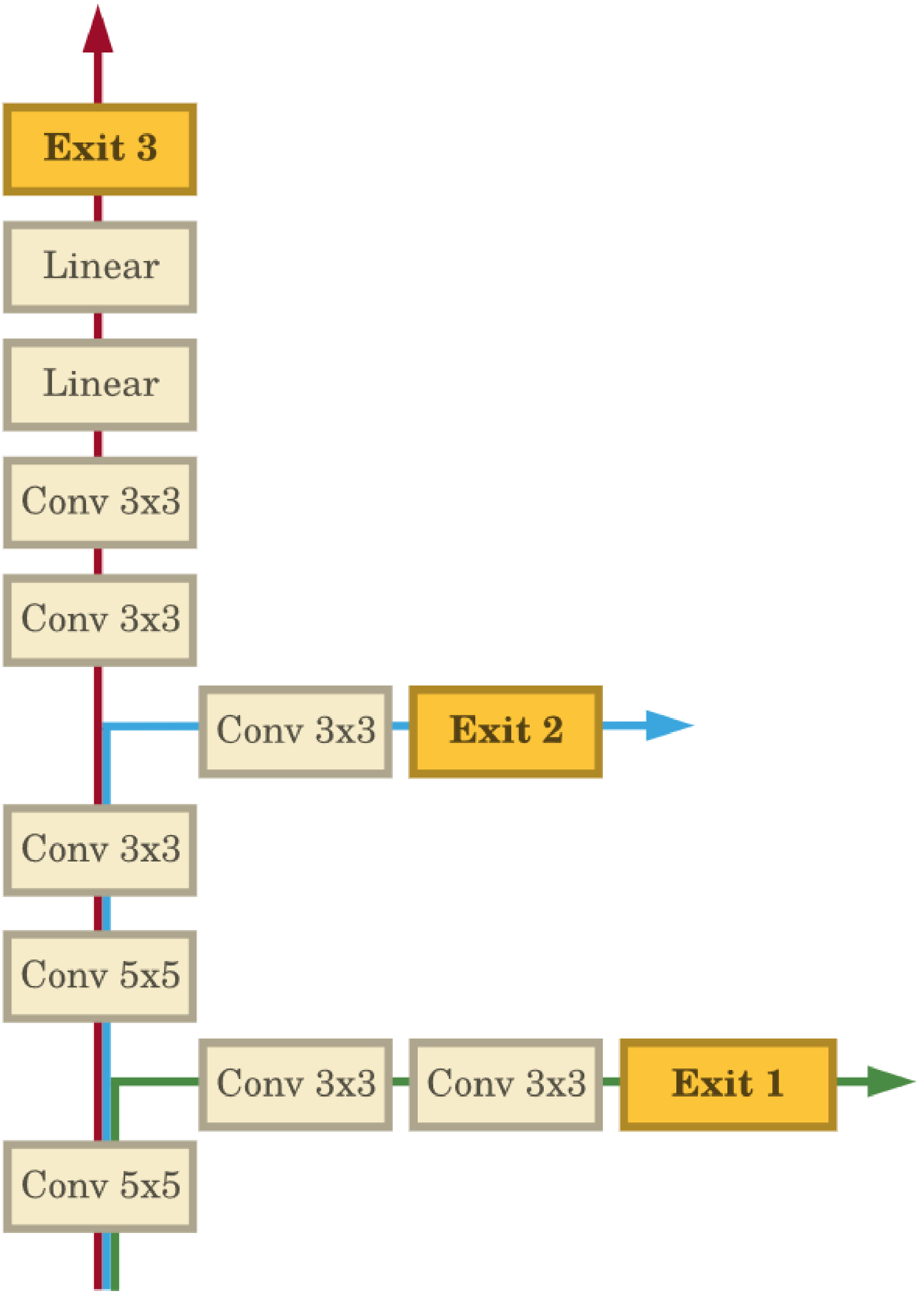


Figure 3.2 BranchyNet [84] was one of the earliest attempts at the Early Exit classifier approach to Neural Networks. It is based on adding two early exit branches to the AlexNet Architecture.

3.4 Accumulating Information on the Values of choices

While ramping activations characteristic of Evidence Accumulation has been found during Value-Based decision making in various animales [37], including primates, much of the research on Evidence Accumulation Models has been focused on perceptual decision-making. In chapter 4, we explore how Evidence Accumulation models can be incoporated into Reinforcement Learning(RL) frameworks. In chapter 5, we explore how Evidence Accumulation can be realized in Echo State Networks trained using RL.

Chapter 4

Reinforcement Learning Linear Ballistic Accumulators

Unlike Supervised Learning algorithms where the learner is directly provided with information as to whether each decision they make is correct or wrong and unlike Unsupervised Learning algorithms where the learner has to learn purely by observing patterns in the input, a large proportion of the skills humans posses are acquired through interactions with the environment [86]. The rewards and punishments experienced as a result of following specific policies are used as cues to learn the optimal actions that needs to be performed [79]. This makes Reinforcement Learning (RL) [83] models in which the agent learns via interacting with the environment, especially suited to study human learning.

It is a well established view that humans make decisions by computing the value of each possible action. [21] This naturally leads us to 2 questions with respect to cognitive science: How are these values computed? and How do these values translate into actions? [102]. The answer to the latter would be an action selection mechanism and in RL models, this mechanism is typically seen as choosing between 2 phases: Exploration, and Exploitation. In the Exploration phase, the model selects actions in a random manner. In this phase, the aim is less about achieving the maximum reward and more about discovering the true underlying value of the states in the environment. The value of a state here meaning the total expected return one can hope to get if they choose the best actions starting from said state(best policy). Once the values of all the states in the environment has been determined, the agent focuses on using the learned values to obtain the maximum rewards in the Exploitation phase.

In RL models, typically this interplay of exploration and exploitation is implemented by a function based the softmax choice rule. However, many of these functions not only biologically implausible but also ”struggle to capture the dynamics of decision making” [63].

Furthermore as these models do not predict Response Times(RT), such models are fit only to choice data. However, the amount of time a subject takes to respond to a choice can be a good indicator of the amount of confidence one has in the action they took [33]. This in turn can be used to estimate the quality the subject perceived of the action. In addition, by not having a comprehensive model of action selection such models fail to account for factors such as response caution and time taken for perceptual processing

This chapter is a slightly modified version of our paper **Modelling Grid Navigation Using Reinforcement Learning Linear Ballistic Accumulators**; Venugopal G, & Bapi R.S. published in IJCNN 2023 and motor action, which could be different across distinct populations and experiment conditions [85] and is also important for taking into account individual differences [64]. In this respect, aside from measures of central tendency such as mean or median RTs, full RT distributions have been hypothesized and shown to capture important properties of mental processes [38].

An alternative to choice rules such as the softmax logistic function are Evidence Accumulation Models (EAM), such as, drift diffusion model(DDM) [70], the linear ballistic accumulator model (LBA) [12], and the leaky competing accumulator model [92]. Such EAMs not only found success in capturing behavior during perceptual decision making and working memory tasks [67], but have also been used to analyse value based decision making [47]. Moreover, the DDM of which the LBA is a mathematically simpler version of, has been shown to be derived from neuronal population dynamics [75].

In race models [94], a sub-class of EAMs, the evidence for each choice in a decision making task is accumulated till one of them reaches a threshold at which that choice is chosen to be executed. The response time of the action is taken to be the sum of the time taken for the evidence to the reach the

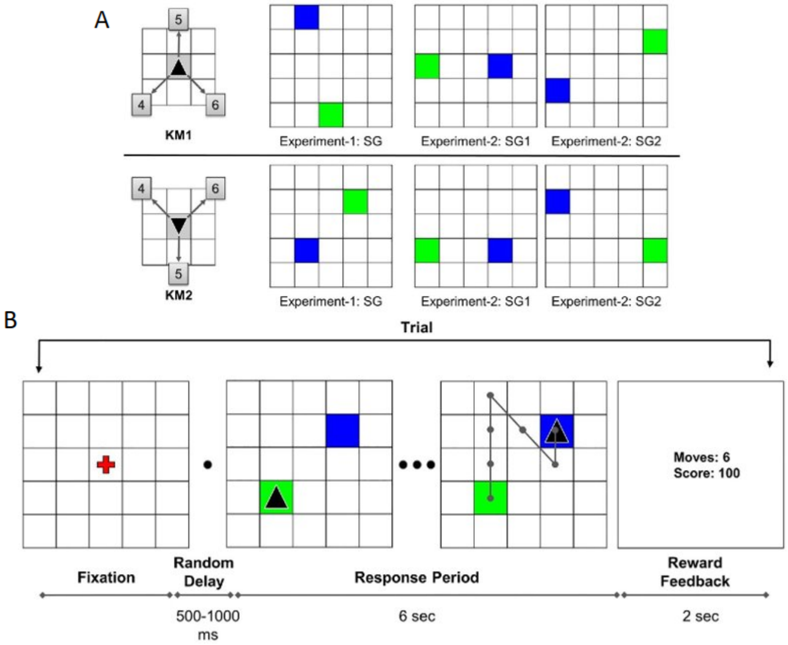
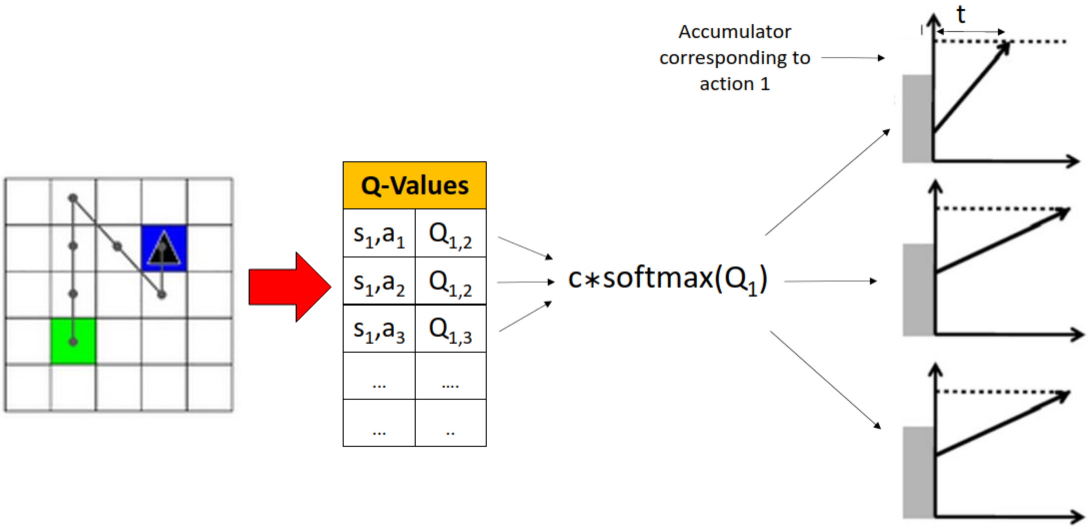


Figure 4.1 Taken from Bera et al. [8](A) Key-mapping (KM) and start-goal (SG) position sets used in the experiment. Each participant was randomly assigned either KM1 or KM2. The boxed numbers on KM figure show corresponding numeric keys associated with the movements. In SG figures, green and blue tiles represent start and goal positions, respectively. (B) Task diagram: sequence of trial events. In this illustration, the participant is assigned key-map KM1. An example optimal trajectory is shown on the grid.

threshold and a non-decision time. The various parameters in such models are related to real world factors. The rate of evidence accumulation for the optimal choice as compared to the other choices is respresentative of the quality of information processing or the subjects' skill in making such choices. While the height of the threshold is representative of the caution one exercises while making the choice and Non-Decision Time is representative of the amount of time spent on perceptual and motor processing. In LBAs, evidence accumulation is represented by a straight line whose starting point is drawn from a uniform distribution, and slope is drawn from a gaussian distribution. The center or mean of this Gaussian distribution is known as the drift rate.

Both RL models and EAM models are complementary to each other. While RL models provide a mechanistic explanation for how information is stored and updated in latent representations in the brain, EAMs focus on how these latent representations are turned into actions.

Thus such RLEAM models have been used to successfully investigate decision making tasks. [63], used RLDDMs to study the effect of stimulant medication on adult ADHD patients. [32], compared RLDDM models with standard DDM and RL models to show that RLDDM models were able to model



Q-Values	
s_1,a_1	$Q_{1,2}$
s_1,a_2	$Q_{1,2}$
s_1,a_3	$Q_{1,3}$
...	....
...	..

Figure 4.2 Illustration of RLLBA Model. Outcomes of actions taken, resulting states and rewards acquired are used to update the Q-values table. For action selection, the Q-values of the current state are transformed by a linking function to be used as drift rates for the accumulator corresponding to each action. While the slope of the line is drawn from a Gaussian distribution centered on the drift rate, the starting point is drawn from a uniform distribution. The threshold, standard deviation of the Gaussian slope distribution and the upper limit of the uniform starting point distribution are parameters that are fitted. The action corresponding to the accumulator which crosses the threshold first is the one that is chosen.

the empirical behavior better. [98], used RLDDM to analyze the behavior of patients with Gambling disorder and also used the RLDDM model fit for a model-based imaging analysis.

However, in all of the above studies, a two armed bandit task is used as the experimental paradigm. Here, we test the effectiveness of such models in capturing behaviour from an internally guided Motor Sequencing Task.

Motor skill learning refers to ”learning a specific subclass of skills that involve sequential motor movements such that they are executed accurately and quickly with practice”. [9] Thus it deals with a wide spectrum of activities ranging from complex tasks such as driving and cycling to simpler ones such as typing and grasping. While the nature of how we learn many such motor control skills makes RL models particularly suited to studying the phenomenon, a large proportion of studies in the domain primarily focus on externally guided sequencing tasks to study Motor Learning [**?**,57]. In such tasks, the move to be taken is directly presented to the subjected and factors like increase in speed and decrease in error rates are analysed over time.

In contrast, in this study we use the Grid-Sailing Task(GST)(Figure 4.1), where participants have to understand the environment and decide which action to take by themselves. Thus, we can say that the Grid Sailing Task is internally guided. When compared to multi armed bandit tasks, in GST, subjects receive rewards depending on the sequence of actions taken as opposed to the reward function being solely based on the previous action. Also, unlike many previous studies using RLEAM models, here the agent has access to upto three possible actions from each state.

Overall, the GST is a well studied task designed to explore the three different stages of motor learning: Exploration, Goal-Guided, and Automatic. Studies have shown that different brain regions are activated during various stages of learning. [31]

4.1 Methods

4.1.1 Experiment

In the Grid Sailing Task [30], the participant is asked to move a cursor located in a grid from a starting position to a goal position within a response period of 6 secs after which the trial will timeout. 3 possible cursor movements are associated with 3 particular keyboard inputs in a one-to-one correspondence. In this particular version of the experiment, the optimal path length (in terms of minimum number of steps) from the start goal to the end goal is 6 steps and there exists multiple optimal paths of this length.

If the participant reaches the goal position in 6 steps, then they receive a reward of 100 points, and for each additional step they take to reach the goal they lose 5 points (e.g., 95 for 7 steps, 90 for 8 steps). If the participant does not reach the goal position, they receive 0 points. The time taken for the participant to press the first key is denoted as the Reaction Time. While the total time taken by the participant for the subsequent key presses is referred to as the Execution Time.

Prior to the start of the experiment, subjects are not aware of the environment keymap,i.e they do not know which corresponds to which movement. The experiment consists of 2 phases. The first phase is designed so that the participant explores the environment to learn the keymap. In computational terms, the idea is to have the subjects explore and learn the underlying model. In this phase, participants are only exposed to single pair of start-goal positions(SG pair). Thus the same SG positions are presented to the participant in each trial.

The second phase is designed to study how the participant would use the model information to gain maximum rewards from the environment. Here, while the same keymap is used, two SG positions distinct from the one in the first phase are used. Thus the subject are forced to plan a new route.

Overall, the Grid-Sailing Task allows us to track the development of voluntary movements from exploration, to goal driven actions to habits effectively.

4.1.2 Model

An environment set up for the RL model to interact with based on the original experimental setup. States were taken according to the position of the cursor relative to that of the goal position(Manhattan Distance). Each state had three possible actions. For the corner cases, only relevant actions would produce a state change.

The reward function was such that the reward given at the end of each step is a linear function of the Manhattan distance between the goal position and the cursor position. Thus the agent would recieve a less negative reward if the action resulted in moving closer to the goal position, as opposed to a more negative reward if it ended up moving farther away.

During action selection(Figure 4.2), the 3 q-values associated with each action the agent could take from the current state are first passed to a softmax function. The inverse temperature of the softmax function is set to 0.35. The output of the softmax funtion is scaled by a scalar parameter, and the resulting values from each Q-value is taken as the drift rate for the LBA model for the corresponding action. As typical, the resulting action and the response time depends on which accumulator croses the threshold first and the intercept the accumulator makes on the threshold respectively. Modulating the threshold as a function of Q-values as in [63] was experimented, however did not give better results while testing simulations. Thus in this study, only the drift rates are varied as a function of RL variables.

The data was fitted to 3 different RL models, a purely model based one and 2 hybrid models. One of the hybrid models used Weight based arbitration [36], while the other used Value of Information(VoI) based arbitration [7]. The model free part was implemented as Q-Learning, while the model based part was implemented via a depth limited value update algorithm. Depth limited search unrolls the model based tree of state, action and resulting state to a pre-specified depth. The state values are obtained by adding the expected returns of actions up to roll-out depth. The updates are then propagated to the root node of the search tree. The Weight based arbitration and VoI based arbitration searched till a Depth of 2 when using model based update, while the Purely Model Based algorithm searched til a depth of 1. These depths were selected based on simulation results(Figure 4.3).

In weighted arbitration, two separate Q-value tables are maintained, one which is updated using Q-Learning after each step, and another which is updated using Depth Limited Search before each step. For action selection, a weighted sum of the two q-values are given to the LBA as below,

$$Q(s,a) = w * Q_{MB}(s,a) + (1-w) * Q_{MF}(s,a), \tag{4.1}$$

where Q_{MB} and Q_{MF} represent the Q-Values tables respectively. The weight, w, is represented by,

$$w = m * e^{-n*t}, \tag{4.2}$$

where m and n are parameters and t represents the trial number.

In VoI based arbitration, model-free and model-based processes are arbitrated via uncertainty based Value of Information given by:

$$VoI(s,a) = C(s,a)/(\alpha(s)+\epsilon), \tag{4.3}$$

where $C(s,a)$ denotes the variance of Q-values corresponding to state s, and action a, over episodes where the state was visited, $\alpha(s)$ denotes the standard deviation of Q-values for different actions from the state s, and ϵ is a constant. VoI based arbitration only uses one set of Q values which are updated using a depth limited search if the VoI is a higher than a certain threshold before making a step and always updated by Q-Learning afterwards.

Since a higher reaction time has been observed in the Empirical data as compared to subsequent key presses as can be seen in Figure 4.4. A constant term which is a parameter is added to the non decision time for the first key press(trial) only. It is interesting to note that in [29], where performance in GST was analysed under various conditions, a pre-start delay was found to increase performance both in experiments designed to study exploratory behaviour and model-based behaviour.

4.1.3 Model Fitting

Model Fitting was done via Maximum Likelihood Estimation. Each action in a trial was simulated 4000 times. Kernel Density Estimation(Seaborn Python Package) was then used to generate probability

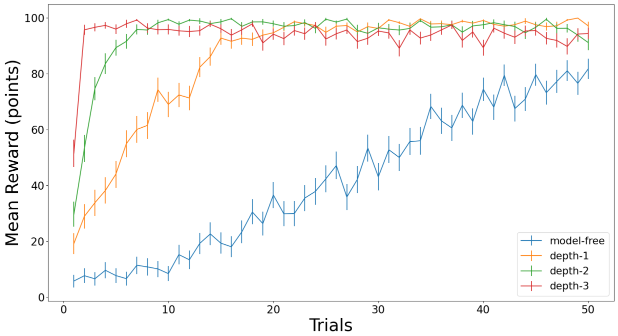


Figure 4.3 A comparison of different RL models. All models are simulated with the same parameters.

density functions of reaction time given the choice. This was used to calculate the sum of negative log likelihoods of each action for each subject. This sum was then minimized using Differential Evolution [48] from the python Scipy package.

The Empirical Data used for fitting is taken from Bera et al. [8]. In the study, subjects first performed the task without knowing the keymap. After the keymap had been learned through exploration, the subjects then performed the same task with the same keymap but with different start goal positions. It was this data that was used for modelling. Models were fit for 11 randomly selected subjects from a total of 42 participants. The data was taken from the Mixed-SG(phase 2) condition. Due to large intra-subject variablities in best fit parameters both when fitting with both conventional RL models and RLLBAs, data from each subject was fitted with a separate model.

4.2 Results

4.2.1 Correlation between Variance of Q-values and Reaction Time

If the variance of Q-values across actions in a particular state is high, this means that the model views the quality of certain actions to be higher than the others, thereby resulting in the agent having lesser uncertainty about which action to choose. Lesser uncertainty results in faster decisions and this is reflected in RLLBAs too, where when variance in Q-values are high, response times are lower.

We tested whether this is true for the empirical data collected from a behavioural experiment. Q-values were updated via 3 different RL algorithms(VoI Based Arbitration, Weight Based Arbitration and Purely Model Based) while following the actions that subjects in the experiment took. We find that there is a significant correlation(pearson correlation, $p - value < 0.001$) for all 11 subjects between Variance in Q-values and Reaction Time when Q-values are updated by VoI based and Weight based Arbitration, but this does not hold for all subjects when a purely model based algorithm is used.

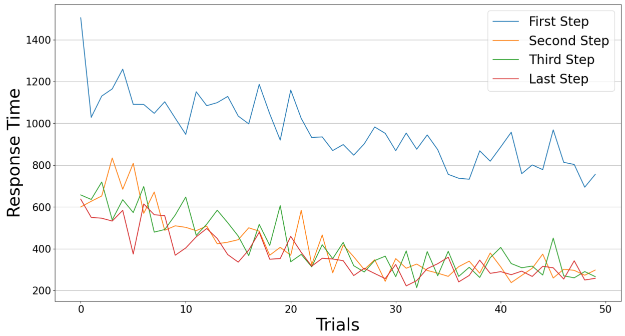


Figure 4.4 Mean Response Times across all 42 subjects for the First, Second, Third and Last steps for the first 50 trials. Human Data taken from [8].

Figure 4.5 shows the scatter plots for Variance of Q-values across response times for a representative subject. The colorbar reprsents manhattan distance between the agent and the goal when the decision is made. We see that while the graphs from VoI based and Weight based Arbitration show clear negative slopes, this is not the case with Pure Model Based updates.

It is interesting to note that the optimal Q-value table would indeed not show much variance as the cost of a sub-optimal step would typically be just one or two more additional steps. This maybe why the points in the scatter plot corresponding to Purely Model Based updating are generally lower in

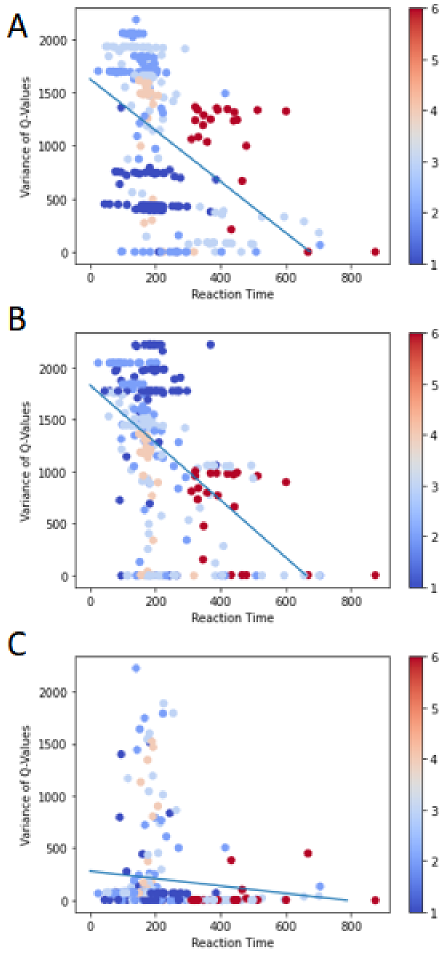


Figure 4.5 Scatter Plot of Variance of Q-Values(y-axis) and Response Time(x-axis). In interacting with the environment the model followed the actions of a subject. The variance in Q-values of the state at which the subject was in before taking the action is compared with the time taken for the subject to respond. (A) Q-Values are updated using VoI-based arbitration (B) Q-values are updated using Weight Based arbitration (C) Q-values are always updated using only model-based updates.

variance. In this way, the arbitration mechanism can be seen as taking cues from the Reaction Time in transitioning from Model Based Learning to Model Free Learning.

4.2.2 RLLBA results matches that from Empirical Data

As can be seen from the Figure 4.7, the graphs obtained from the simulations match those from the computational model remarkably well. It is worth noting that in the experiment, participants were explicitly advised to maximally re-use the explored trajectories in order to execute the task quickly and accurately. This was not incorporated into the computational model in any way, but it may explain the slightly higher reaction times of the simulations to the end of the task as compared to the empirical data.

We also simulated updating the Q-values with VoI based arbitration while following the actions of subjects. In Figure 4.6, the average probability of predicting the correct action across each episode

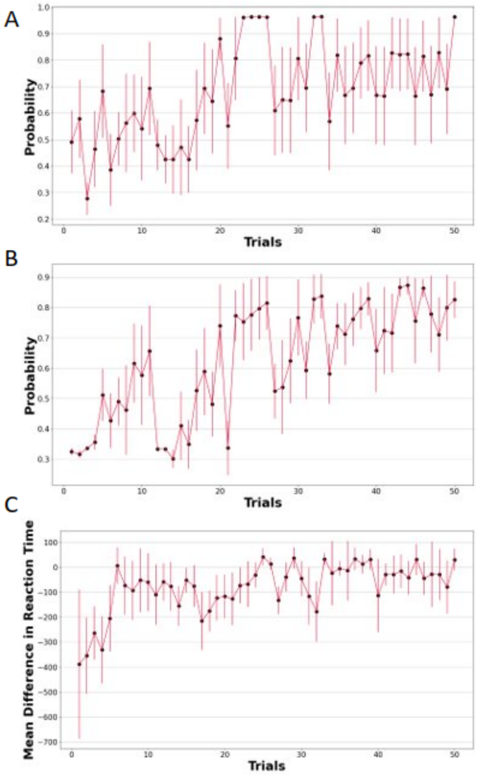


Figure 4.6 Probability of Predicting the action that the subject took, and the difference in response time predicted by the model and the time that the subject took is averaged across each step for each trial. (A) Probability of correct prediction by VoI arbitrated RLLBA mddel (B) Probability of correct prediction by conventional RL model (C) Mean Difference in RT for each trial. Here the difference is calculated by subtracting the time taken by the subject from time predicted by the RLLBA model.

along with the average difference in Response Time that is predicted by the model and that was taken by the subject is plotted. When the Q-values are fed to the LBA, the accumulator model is able to predict the action that the subject took with a mean probability of 0.65, which is significantly above chance probability.

It should be noted that the likelihood function that was maximized while fitting the model was a function of both the probability of taking the ground truth action(action taken by the subject at the particular episode) and the probability of LBA producing the ground truth reaction time. In Figure-4.6(C), we see that the RTs predicted by the model are consistently lower than those taken by the subject. While the magnitude of difference is different across subjects, the negative difference in RTs are a consistent property. In the context of how the likelihood function is calculated this result can be interpreted as the model not being able to produce appropriately high RT without sacrificing probability. More about this is expanded in the Discussion section(Section D).

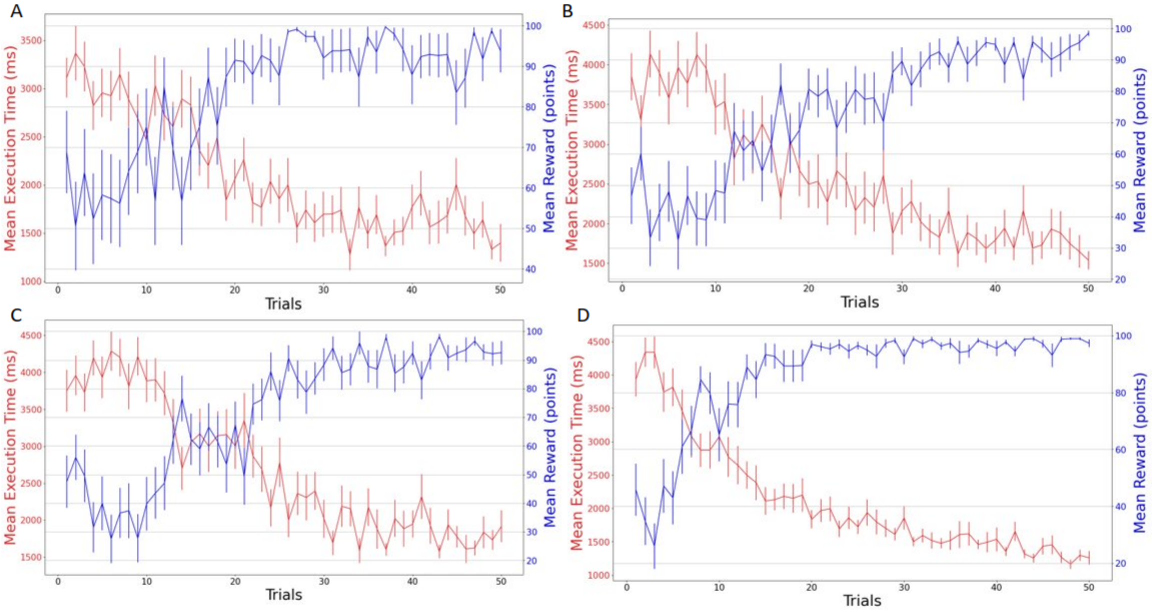


Figure 4.7 Trial-by-trial course of change in response times. The bars on the plot data-points denote standard error in measurement.(A)Mean Execution Time and Mean Reward obtained averaged over all participants selected for model fit.(Empirical Data) (B)Mean Execution time and Mean Reward averaged over simulations of VoI based arbitration model(depth-2). (C)Mean Execution time and Mean Reward averaged over simulations of Weight based arbitration model(depth-2).(D)Mean Execution time and Mean Reward averaged over simulations of Purely Model Based RL(depth-2). The model simulations consist of 1 simulation per subject using the best fit parameters.

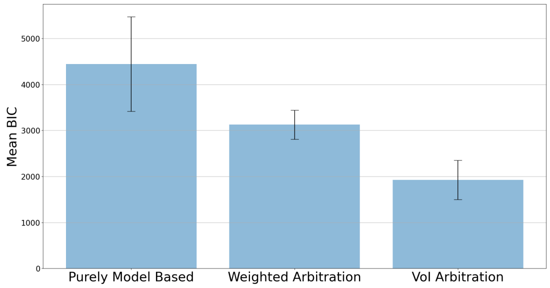


Figure 4.8 BIC values for each model. Mean of the BIC values are taken across all 12 subjects. The error bars denote the standard error in measurement(SEM).

4.2.3 Bayesian Information Criterion(BIC)

Bayesian Information Criterion(BIC) [80] values were calculated for each of the three models. The values are given in Figure-4.8. Lesser BIC values indicate better fits.

While the best arbitration scheme for conventional RL models was found to be Weighted Arbitration [8], when combined with LBA we find that the best scheme is VoI based arbitration. This showcases how the inclusion of response time information can give rise to alternative interpretations.

4.3 Discussion

We see that RL models are able to capture human performance from the Grid Sailing Task remarkably well. The VoI based models are able to predict the action that the subject takes with high probability, and while also predicting response times within a reasonable degree of error.

4.3.1 Using RL when choices are not forced

RL has found significant success in modeling human behaviour in forced choice action tasks. A sub-class of such models, the two-stage task has been extensively used to study the interaction between goal-oriented and habitual processes in the brain. However, in many real world tasks, the consequences of an incorrect action can be quite high. In such tasks, it can make sense for the model to wait for a certain period of time before making an action.

In the Machine Learning domain, while for specific tasks, complicated mechanisms of choosing not to take an action have been proposed, the most common way of accounting for this dilemma is by including a ’No-Go’ choice. However, this requires the policy to actively choose not to act, which is counter-intuitive. For comparison, in accumulation models the default choice is to not act, and an action will only taken when the evidence crosses a threshold. [1] reports that an A2C-RNN model combined with an accumulator showcased better performance than a state-of-the-art deep reinforcement learning models in a mode estimation task.

In the Grid-Sailing task, participants are not forced to make a decision. At times, a participant may choose to spend time thinking and may only make 4 or 5 moves during the response period, another subject might choose to explore more aggressively and make 10 or 11 moves. Thus to model the timeout in the task by setting an arbitrary limit on the number of steps the agent can take in a trial might be the best choice. Once again, in cases such as this, integrating EAMs into RL algorithms come across as an intuitive way of solving this problem. Here we showcase that indeed, such models are effective at capturing behaviour in such situations.

4.3.2 EAMS as a model for exploration in the brain

It is interesting to note that several brain areas where neural correlates of EAMs have been found have also been theorized to play a role in RL. For example, parts of the striatum have been implicated in learning action outcome associations [44] and in setting response caution [93]. Similarly, the parietal cortex has been implicated in perceptual evidence accumulation [81] and state prediction error [36]. In this context, it is worth to note that neural mechanisms behind exploration is still a topic of study today in cognitive science [14], and EAMs in RLEAMs not only handle the speed accuracy tradeoff but also the exploration exploitation tradeoff. This motivates a mechanism for action selection in the brain based on competition and evidence accumulation and models of this kind are already being proposed and studied [27].

RLEAMs can help decompose mechanisms of choice and learning in a richer way than could be accomplished by either RL or DDM models alone, while also laying the groundwork to further investigate the neural underpinnings of these subprocesses by fitting model parameters based on neural regressors. In this way, this study makes a significant contribution to the work bridging theories of decision making and RL.

4.3.3 Incorporates more factors into action selection

EAMs allow us to ask more nuanced questions. For example, in [69], the authors model human performance data in a lexical decision making task across two groups, young and aged. While they found that response times of the subjects in the aged group responded were significantly slower than those in the young group, there was no significant difference in the rates of evidence accumulation in the two groups. Instead the older participants, showed an increased non-decision time and increased response caution respresented in the model as higher thresholds.

While RL models focus on how information is stored and updated, EAMs focus on how this is translated into action. As such, not only do RLEAMs allow for decoupling of group effects into whether the effect is due to a problem in learning or a problem in action selection, they also allow us to answer more nuanced questions, such as, whether people who exploit more, are also more cautious in making decisions.

4.3.4 Usefulness of RT Distributions in determining Arbitraton Mechanisms

A major line of research in Cognitive Science is studying the interaction between goal-directed systems and habitual systems. Numerous studies have suggested that humans use a combination of both these systems in executing various tasks, and that different parts of the brain are involved in these processes; the Dorsolateral Prefrontal Cortex for Model-Based Learning [22, 101] and the Posterior Dorsal Striatum for Model-Free Learning [89, 101]. Due to many behavioural features shown by these two learning systems in situations such as outcome devaluation [39], goal-directed systems are commonly modelled using Model-Based RL while habitual systems are modelled using Model-Free RL.

Two popular ideas of arbitrating between these two systems are Weight Based Arbitration [36] and VoI based Arbitration [46] [97]. While both methods make different predictions about the mechanisms of arbitration in the brain, when fitted to behavioural data the difference in quality of fits in these models are not high enough to make definitive conclusions, and often this is the case even if the analyses are supplemented with neuroimaging data [51].

We see that in the case of RLLBAs however, there is a clear distinction in the BIC values for different arbitration mechanisms. While this can be attributed, cues taken from RTs, it should be noted that RT distributions can also play a major role. For instance, even in the final trials of the experiment, subjects have shown significant variation in RT. For example, the second step might be taken significantly faster than the third step. In such cases it should be noted that higher drift rates result in sharper peaks, and the model may be forced to further finetune its parameters.

4.4 Conclusions

There are many cases in which we may make sub-optimal decisions, even when we may be perfectly capable of performing optimally [61]. The system which translates the latent representations in the brain to action has its own dynamics and can be affected by multiple factors. Reinforcement Learning Evidence Accumulation Models provide a better model than Reinforcement Learning or Evidence Accumulation Models individually to decompose and analyse effects of various conditions.

In this study, we show that such models are able to capture human behaviour in a internally guided motor sequencing task effectively. Not only that the provide clear distinguishable fits to 3 different RL algorithms, further supporting their usage to model human behaviour when asking more nuanced questions.

Further study can look into architectural changes to the model such as having different learning rates for positive and negative prediction errors [35] or using time varying boundaries [104].

Chapter 5

Exploring the structure of the Basal Ganglia using Echo State Networks

5.1 Introduction

The structure of the basal ganglia is remarkably similar across a number of species, from the newt to the primate [11]. These ganglia are often described in terms of pathways, including the direct, indirect and hyperdirect pathways. The role of each pathways is still under scrutiny and consequently, there exist several hypothesis regarding their role in action selection and decision making for which basal ganglia are known to be deeply involved. In this article, we are primarily interested in exploring the role of structure when solving a decision task while avoiding to make any strong assumption regarding the actual structure. To do so, we exploit the echo state network paradigm that allows to solve complex task based on a random architecture [41]. Considering a temporal decision task, the question is whether a specific structure allows for better performance and if so, whether this structure share some similarity with the basal ganglia. Unfortunately, we cannot explore each and every variant of architecture because the number of different structures for a fixed number n of neurons is huge (and grows exponentially with n). Instead, we restrict our exploration to a much smaller subset where a model is made of one to three ganglia with different connectivity and overall organization. We also added a specific continuous case based on topological reservoir that allows to have distance based connectivity patterns and allows us to extend the one ganglia structure. These models are loosely inspired from the direct and hyperdirect pathways of the BG [76], with the latter allowing the production of a fast ”stop signal” thanks to a reactive inhibition.

5.1.1 Structured ESN

The role of structure in ESN have already been addressed in a number of works. While [19] quantified how structure affects the behavioral characteristics of the ESN, several studies have demonstrated that replacing the initial random topology of the ESN by more organized structures could improve the overall performances of the model. Nonetheless, rather than completely removing the randomness of the network topology, certain structures allows to combine both random and structured connections.

One well-known example is the small world network, which has been observed in the neural network of the C. Elegans [96]. Small-world network has also been identified in other brain systems [6], and [4,15] have shown that incorporating small-world structure into ESNs results in performance improvements on benchmark tasks. Various other structures for ESN have also been studied, including the combination of scale-free and small-world networks which demonstrated significantly superior performance [23,43,45]. Additionally, modular structures [73], forward topology with shortcuts pathways [26] and hierarchically clustered ESNs [42] have been explored, each impacting memory capacity, temporal properties, and reservoir stability. An alternative approach involves investigating various structures by combining multiple random reservoirs instead of a single one. This method known as Deep Reservoir Computing introduces richer temporal dynamics in the models [34,56]. In the study of [103], multiple reservoirs are combined thanks to lateral inhibition. Although these various studies demonstrate the impact of ESN topology on performance, the work by [49] stands out as the sole study indicating that no other topologies exhibit significantly superior performance than random ones. Emphasizing this particular finding is one of the aspect we want to highlight.

5.1.2 Time-constrained decision

In decision-making, the temporal aspect is a significant component taken into account in the current decision-making task. In the real world, decision Making is time constrained. Decisions need to be taken within certain timeframes, where the importance of speed and the need for caution can vary across situations. In many such cases, there would exist a speed-accuracy tradeoff, where one can collect more information or ponder more over the choice in order to make a better decision at the cost of taking more time. As navigating such trade-offs optimally would be important for one's survival(for eg., [20]), many sophisticated models have been developed to model animal behavior in such situations. A popular set of models used to study how animals approach time-constrained decisions are Evidence Accumulator Models(EAMs) [13,66]. In such models, deliberating over a decision is modeled as accumulating observations which over time which can be perceived as evidence for making one or another choice. When the accumulated evidence reaches a certain threshold, a decision is taken. Such models are able to seamlessly integrate the myriads of factors that affect animal decisions. The height of the threshold represents response caution, whereas each observation obtained can be analyzed as probabilistic likelihood for taking one choice or the other. Furthermore, electrophysiological evidence for ramping signals correlated with evidence accumulation has been found in certain regions of the brain([65]). Finally, widely used EAMs such as Drift Diffusion Models are equivalent to the Wald Sequential Probability Ratio Test, a widely used to method to make decisions in Signal Detection and Estimation Theory [68]. Despite the many factors in favor of EAMs as a model of decision making. One occasion where such models fail is when new evidence comes into light and the decision needs to be changed quickly [17]. As EAMs factor in all observations since decision onset, it is harder for them to respond quickly to sudden changes especially in cases where the new evidence is contrary to previously received evidence. While alternative models such as Leaky Accumulator Models [91] and Urgency Gating Models [87] have been proposed as a solution to this problem, they often don't provide as good fits to animal behaviour when considered across a wide variety of tasks. This paper introduces an alternative approach to address this challenge. The objective is to construct Echo State Networks (ESNs) with multiple architectures, where each distinct component of the structure is capable of handling temporal information in a different manner.

5.2 Methods

5.2.1 Tasks

We consider a non stochastic two-arm bandit task where an agent is presented with two options, each associated with a certain amount of reward. Once a choice is made, the agent receives a reward based on the amount attached to the chosen option. This trivial task is further complexified by introducting a choice indirection (motor aspect), different encoding (spatial aspect) and differential timing (temporal aspect).

An example is depicted in 5.3 where each option corresponds to a specific cue shape (square, round, lozange, triangle). Once the agent has made its choice, it receives a reward based on a value associated with the selected option. The task is an evolved 2-arm bandit task with the introduction of both time aspect and position aspect.

Motor aspect An option is represented by a stimulus with a given identity (1 to 4) and a given position (1 to 4). For a given trial, stimuli identity and position are mutually exclusive. The value of an option is solely attached to the identity of the stimulus, irrespective of its position. The agent's choice is interpreted as a position from which the identity of the simulus can be retrieved (and hence the amount of reward).

Spatial aspect We considered two different strategies for encoding an input. One (bound) is based on the Cartesian product of position and identity, resulting in a 4×4 input matrix, the other (unbound) is based on the separate representation of identity (vector of size 4) and position (vector of size 4). Both strategies allows to encode two mutually exclusive options (i.e., different identity and different position). However, only the first strategy allows to identify each option (solving the binding problem) while the second strategy is ambiguous.

Temporal aspect The onset and offset times of the two options are independent. This means that they may or may not have the same duration, they may or may not start nor finish at the same time and they can be completely disjoint, i.e. there is no overlap between the two options. This temporal aspect considerably complexifies the task. In some trials, the agent must maintain the value of the first option (working memory) while in other trials, the agent has to deal with a late but better option (time constrained decision).

Time aspect The apparition of the two options follows a timeline depicted in Figure 5.2. The chronogram represents two distinct input channels, corresponding to the activation of two cues during a single trial. The parameters d_1 and d_2 denote the duration of these input signals, while $t_2^+ - t_1^+$ corresponds to the time delay between the activation of these cues. Each trial starts at an initial time denoted t^+, and ends at a time denoted t^-. The agent finally receives a reward at a specific time t_{reward}. The two options can either become active simultaneously, where $t_2^+ - t_1^+ = 0$, or with a delay, where $t_2^+ - t_1^+ > 0$. Both options maintain a consistent reward probability of $p_R = 1$, but the reward can take different values among $[0.2, 0.5, 0.75, 1]$. This temporal aspect introduces complexity to the task since the two cues may not appear simultaneously and may not persist for the same duration. In certain trials, the agent must possess sufficient working memory to recall the value of the first cue. Additionally, in trials where the second cue emerges late and is visible for a brief period, the agent must react promptly.

Position aspect: the binding problem. During each trial, the two options appear in two of four possible locations: up, down, right, or left, as depicted in Figure 5.3. The agent has to select the position associated with the most rewarding cue. This type of binding problem has been studied in [88], where the author built a model composed of three types segregated circuits including an associative one allowing to link both cognitive and motor decisions. In total, there are 4 distinct cues and 4 different positions, resulting in a total of 72 possible cue-position combinations. In this study, two input representations are employed: the binding problem is either explicitly addressed using a four by four matrix (Figure 5.4-Left), or it remains ambiguous with a two by four matrix (Figure 5.4-right). In the first scenario, each activated cue is explicitly associated with a position. In the second scenario, this association is ambiguous. In this latter configuration, disambiguating the task relies solely on the timing of cue appearances: if the activation of the first cue does not align with the activation of the second cue, disambiguating the problem becomes possible.

5.2.2 Models

An Echo State Network (ESN) is a specific type of reservoir computing [41] that is a recurrent neural network composed of randomly connected units, associated with an input and an output layer. Only the output neurons, referred as the readout neurons are trained, as depicted with the red arrow in Figure 5.1. The neurons have the following dynamics:

$$\frac{1}{\alpha}\frac{d\mathbf{x}}{dt} = -\mathbf{x} + \tanh(W.\mathbf{x} + W_{in}.\mathbf{u} + W_{fb}.\mathbf{y}) \tag{5.1}$$

$$\mathbf{y} = W_{out}.\mathbf{x} \tag{5.2}$$

where $\mathbf{x}$, $\mathbf{u}$ and $\mathbf{y}$ represent the reservoir states, input, and output. W, W_{in}, and W_{out},W_{fb} are weight matrices, while $tanh$ refers to the hyperbolic tangent function. α refers to the leak rate, a crucial parameter of the ESN that plays a central role in controlling the memory and timescale of the network's dynamics: a small leak rate indicates a bigger memory and a slower dynamics, whereas a big leak rates lead to a smaller memory but a higher speed of update dynamics [50]. All models have been implemented using the Python library ReservoirPy [90].

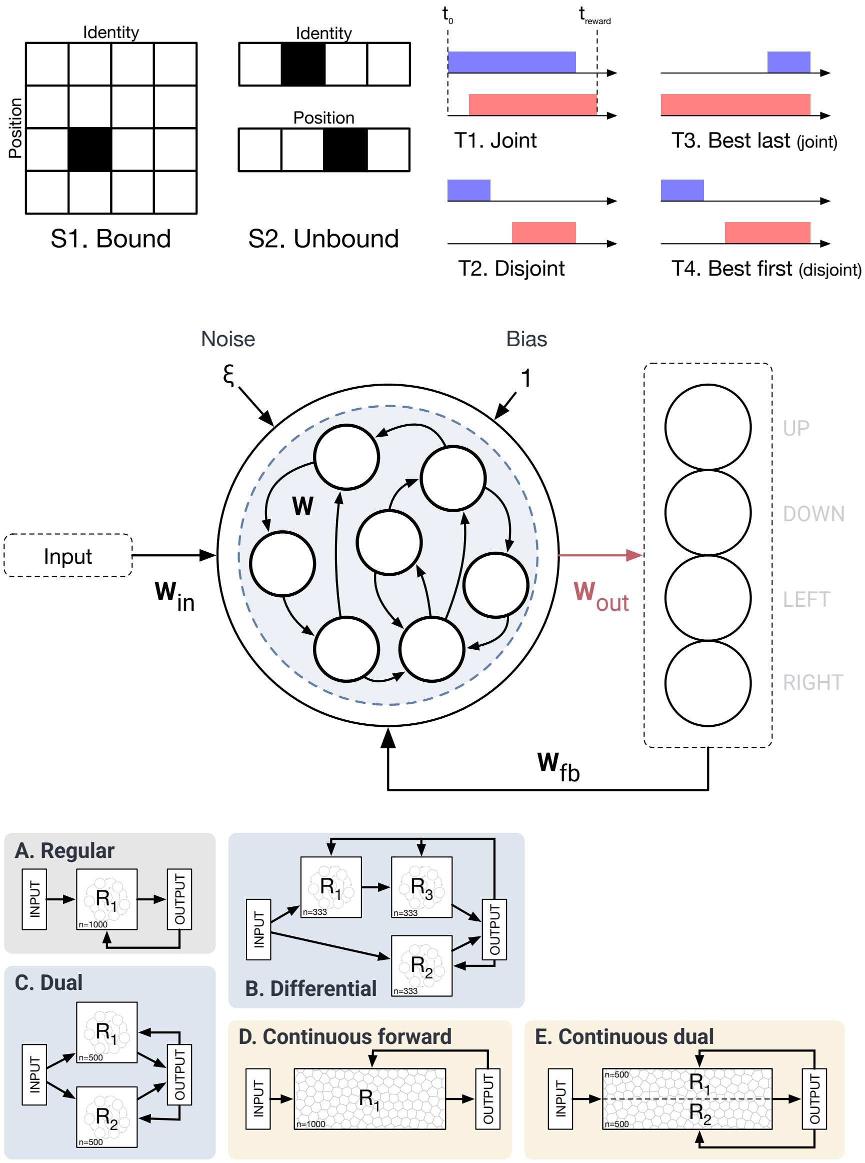


Figure 5.1 Top left: S.1-bound strategy, 4 × 4 input matrix adequate for resolving the binding problem. The rows represent the positions and the columns represent the identity. S.2-unbound strategy, one vector for the position and one vector for the identity. In both examples the activated stimulus is identity 1 at position 2. **Top right**: The red and blue stimuli can have different duration and appearance different timings. The reward always emerges at the end of the last cue. They are joint when their appearance overlap in time, disjoint when they don't overlap. **Middle**: Model architecture with a motor output (direction of movement). The black arrows are fixed and the red are plastic. **Bottom**: Model structures composed of one or several ESN, all connected to the readout a receiving a feedback from it. A-Regular ESN. B, and C are composed of several ESNs constituting two distinct pathways. D is a single ESN with random distance-based internal connections. H is a single ESN composed of having differential connectivity patterns in the upper and lower parts.

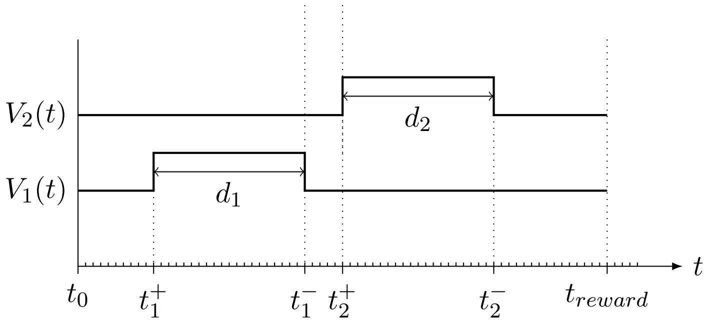


Figure 5.2 Task chronogram. The two stimuli V_i are characterized by their respective onset (t_i^+) and offset (t_i^-) time. The time of decision t_{reward} is fixed and constant across trials.

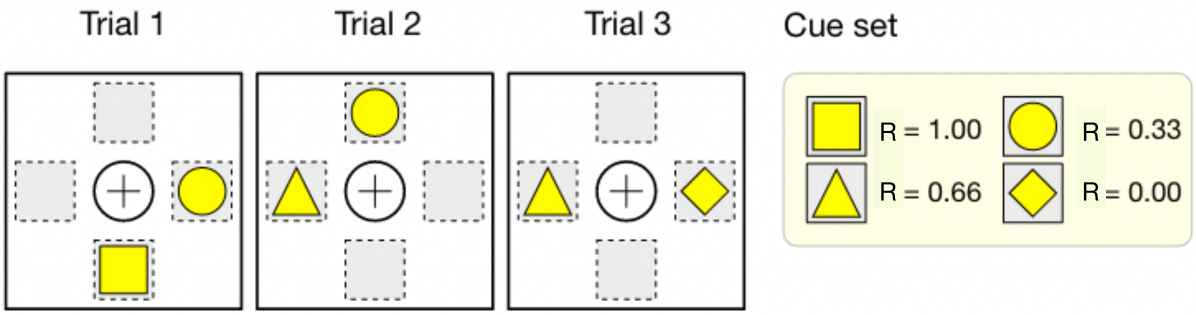


Figure 5.3 Binding problem: the agent has to choose the position associated with the most rewarding cue.

5.2.2.1 Architecture

From the classical ESN (figure 5.1A), we derived several architectures (figures 5.1B,5.1C,5.1D) that are all characterized by the presence of two distinct pathways. the slow pathway, composed of one or two reservoirs, and the fast pathway composed of one reservoir. This draws inspiration from the direct and hyperdirect pathways of the basal ganglia [76], with the latter allowing the production of a fast ”stop signal” thanks to a reactive inhibition. This ”stop emergency brake” [3] is attributed to the significant role of the nucleus STN. All reservoirs are connected to the readout and receive feedback signals from it. Both of these two pathways can receive the input as a whole (option 1 + option 2) or as split input. That is, each pathway receives a single option. More precisely, the slow pathway receives the earliest option and the fast pathway receives the latest option.

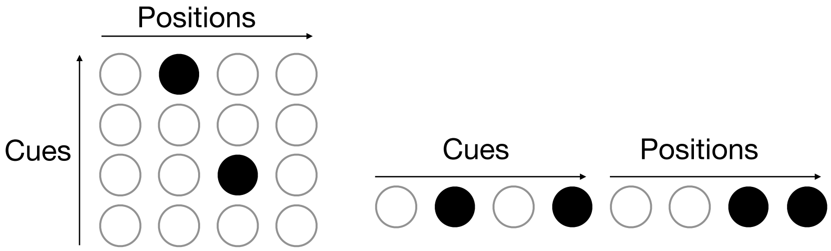


Figure 5.4 Input Format. Left: a four by four matrix adequate for resolving the binding problem. Right: a two by four matrix making the representation ambiguous and impossible to discern which cue is associated with which position.

We also designed topological reservoirs (figures 5.1D and 5.1E) that are reservoirs equipped with a topology [74]. In such reservoirs, it is possible to constrain activity propagation along a feed-forward axis (from input to output). This allows the reservoir to progressively process information along the main axis, where early units (that are closer to the input) have access to local and recent information while late units have access to global and processed information. The output layer which has access to both early and late units has the ability to accumulate information and take accurate decisions, while at the same time having the ability to quickly respond to changes in the environment. To make these type of reservoir, the distribution neurons across a 2D space is first defined by using a the algorithm described in [74] from which the connectivity matrix can be derived. Individual connections are established based on the nearest neurons that meet angle constraints as shown in figure **??**, connections are established between input and output neurons following a rule in which the probability of connection exponentially decreases with distance.

The major difference between the Split models and the original ones is that they process the input signal differently: in the Differential and Dual models, both pathways are fed by the two activated stimuli, V_1 and V_2 of Figure 5.2. In the Split Differential and Split Dual models, the earliest cue V_1 is fed into the slow pathway, and the latest cue V_2 is fed into the fast pathway. Consequently, the Split Differential and Split Dual models processes the two cues separately through its two pathways. These models draw inspiration from the direct and hyperdirect pathways of the BG

All models are based on the same ESN framework but differ on the internal architecture, i.e. the pool of neurons represented in blue in Figure 5.1 vary across the different models.

The first type of topology involves constructing a model with multiple reservoirs [34] as the Dual, Split Dual, Differential model and the Split Differential model illustrated in Figure 5.1. These models have identical overall architectures, consisting of two or three reservoirs forming two distinct pathways: the ”slow pathway”, composed of one or two reservoirs, and the ”fast pathway” composed of one reservoir. All reservoirs are connected to the readout and receive feedback signals from it. The major difference between the Split models and the original ones is that they process the input signal differently: in the Differential and Dual models, both pathways are fed by the two activated stimuli, V_1 and V_2 of Figure 5.2. In the Split Differential and Split Dual models, the earliest cue V_1 is fed into the slow pathway, and the latest cue V_2 is fed into the fast pathway. Consequently, the Split Differential and Split Dual models processes the two cues separately through its two pathways. These models draw inspiration from the direct and hyperdirect pathways of the BG [76], with the latter allowing the production of a fast ”stop signal” thanks to a reactive inhibition. This ”stop emergency brake” [3] is attributed to the significant role of the nucleus STN.

The second type of topology involves single reservoir models as the Continuous forward model and Continuous dual model depicted in Figure 5.1. In such reservoirs, the notion of depth is within the reservoir itself: earlier layers would represent less processed recent information, whereas later layers would represent more processed information representative of a larger timeframe. An output layer which has access to both early and later layers would have the ability to accumulate information and take accurate decisions, while at the same time having the ability to quickly respond to changes in the environment. To make these type of reservoir, the distribution neurons across a 2D space is first defined by using a stippling algorithm [74] that transforms an image into a density distribution in xy-space, where the density at any region correlates with darkness/shade of that area in the image. It has been shown that difference in densities can affect with information processing, such as neurons in high density areas representing more integrated and complex functions than in low density areas. Then, neuron connections are established by confining all links to the nearest neurons that meet angle constraints as depicted in Figure 5.5, connections are established between input and output neurons following a rule in which the probability of connection exponentially decreases with distance.

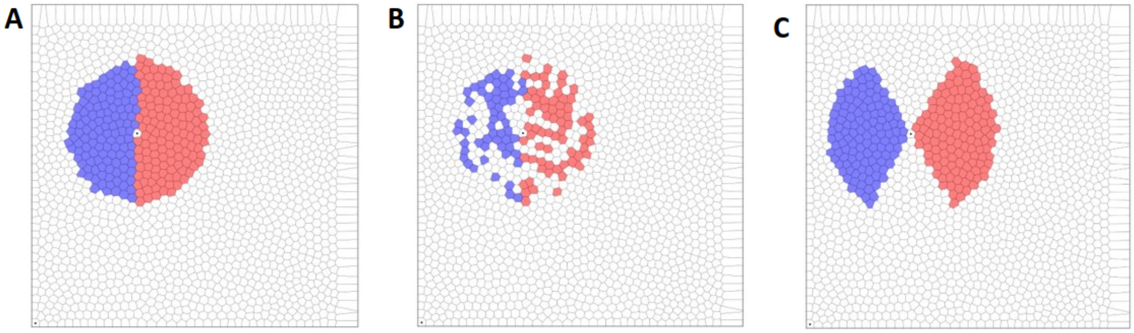


Figure 5.5 Effects of change in parameters on connectivity patterns in Forward Direction models. (A) Connection Pattern when connection angle is set to 90 degrees and connection probability is 1. (B) Connectivity when connection probability is reduced to 0.5 while angle is not changes. (C) Connections when the connection angles is reduced to 75 degrees and connection probability is 1.

5.2.2.2 Learning

The readout layer is trained using online reinforcement learning (RL) based on equation 5.3 and 5.4, where only the weights associated with the selected choice undergo updates. The choice of RL as the learning rule comes from its biological plausibility, given that cortico-basal-ganglia (BG) circuits are trained through reinforcement, thanks to the encoding of reward prediction error (RPE) with dopamine [5]. Equations read:

$$W_{out}(choice) = W_{out}(choice) + \delta W_{out} \tag{5.3}$$

$$\delta W_{out} = \eta.(r - softmax(\mathbf{y}, \beta)[choice]).(\mathbf{x} - \mathbf{x_{th}}) \tag{5.4}$$

where $choice$ represents the index associated with the model's chosen action. η is the learning rate. The function $softmax(y, \beta)$ applies softmax to the model's output with y as the model's output and β as a parameter. x_{th} denotes a small constant value, and r corresponds to reward feedback received. The action selection process follows the epsilon-greedy method, allowing to balance between exploitation and exploration phases. When the agent is in the exploitation mode, it selects the action that corresponds to the highest output value of the model ($argmax(output_{model})$). In contrast, during exploration, the agent randomly selects one action from the set of all available actions, with equal probability among the four possible choices. The method uses a parameter called epsilon (ϵ), which starts at 1 during the beginning of each simulation and ends at 0, signaling a shift towards exclusive exploitation of learned knowledge. This dynamic ϵ adjustment enables the agent to transition from exploration to exploitation.

5.2.3 Protocols

Model optimization. All models underwent a hyperparameter optimization process using the Optuna library [2]. The model parameters such as the spectral radius (sr), leak rate (α), the input connectivity, the output connectivity, the reservoir connectivity, the exploration rate, η and β were optimized. The optimization consists of 600 simulations with different set of parameters sampled using Tree-structured Parzen Estimator (TPE) [10]. Each simulation consists of 1000 trials, and performance assessment occurs over the last 200 trials of the simulation by counting the number of successful actions (best option chosen). The models underwent optimization with task parameters configured identically to the temporal generalization tests (see below).

Temporal generalization tests. The models were optimized across a broad spectrum of timing and delay conditions. More specifically, while $t_{t_1^+}$ and t_{reward} are fixed and respectively set to 5 and 1, d_1 and d_2 vary within the range of 5 to 20, and $t_1^+ - t_2^+$ fluctuates between 0 and 20, with these values being randomly generated from trial to trial. This approach enables to identify the optimal parameters that yield superior performance across all potential delay scenarios. This performance assessment is designed to quantify the extent to which the models demonstrate temporal task generalization capabilities.

No delay scenario. Following the optimization of the models for the temporal generalization test, they undergo evaluation under conditions with no delay between the cues ($t_1^+ - t_2^+ = 0$). Fixed values for d_1 and d_2 are maintained, both set to 5. This assessment offers insights into performance in the most basic case.

5.3 Results

All models were tested after training on 10 different seeds, with the success percentage representing the proportion of correct choices out of 1000 trials. While the overall performances are consistently displayed in blue, the results are further studies by separating two scenarios: when the best stimuli appears first and when the best stimuli appears last. This separation provides a more detailed understanding of the trials in which the models performed optimally.

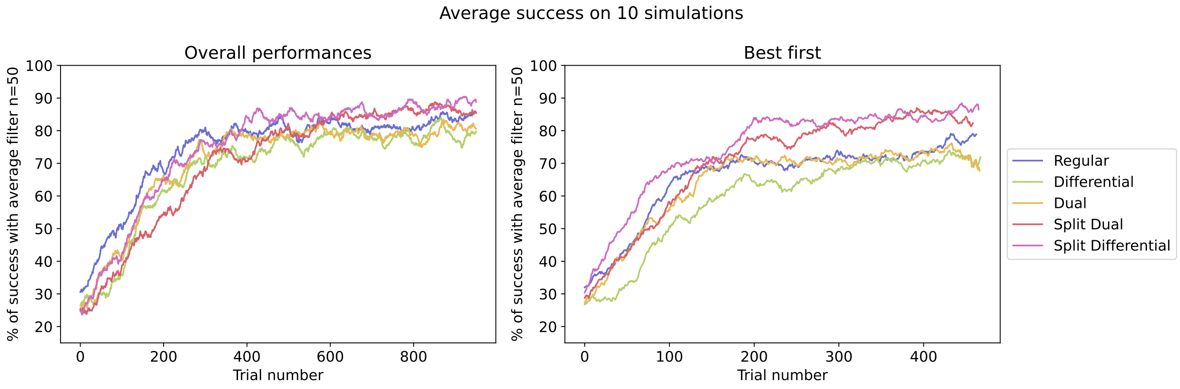


Figure 5.6 Training process. Left: at the end of the training, all models exhibit similar overall performances. **Right**: the superior performance of The Split Differential and the Split Dual when the best cue appears first can be observed during training.

5.3.1 Multiple reservoirs models

5.3.1.1 Input with 4 by 4 matrix

The results illustrated in Figure 5.6 indicate that, while the Differential, Dual models exhibit slightly lower performance compared to the Regular model (respectively 77.5%, 81% and 84.5%, shown in blue), the Split Differential and Split Dual models outperform the Regular model, and this significantly for the Split Differential with an overall performance of 89.8%. This performance boost is primarily attributed to enhanced performance during trials when the best cue appears first (green bar in in Figure **??**), where the performances increase for the Split Differential model and the Split Dual model is around 10% higher. This improvement in performance is visible in the training process depicted in Figure 5.6: the Split Differential model and the Split Dual models in red and pink outperform the other models when the best cue appears first (down Figure), while demonstrating comparable performances in overall

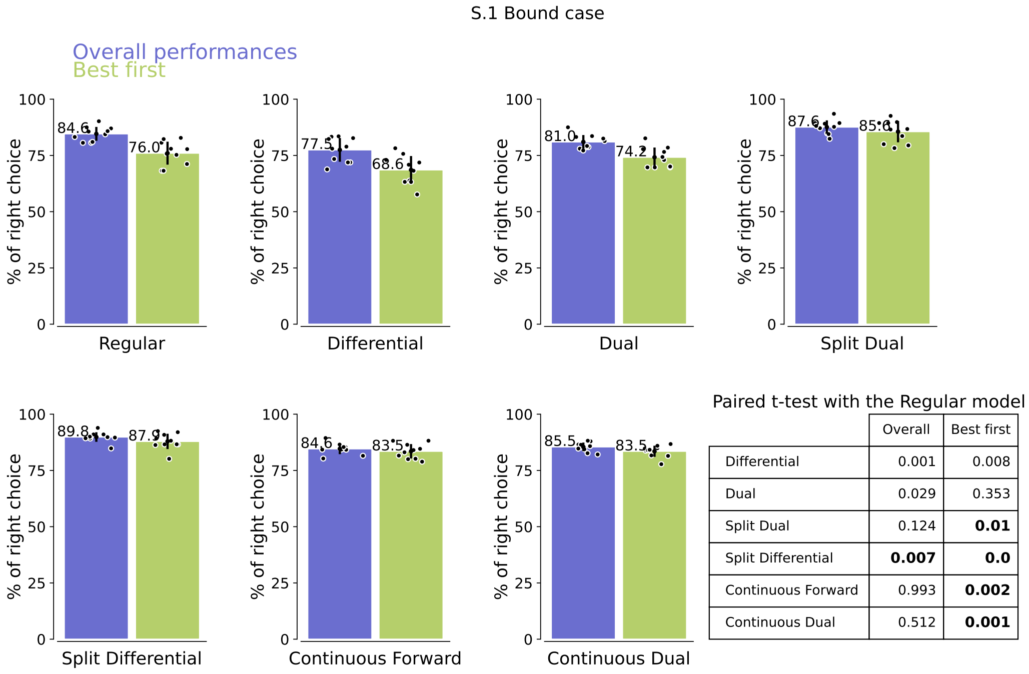
Leak rate values (α)					
Regular	**Dual**		**Differential**		
R1	R1	R2	R1	R2	R3
0.070	0.060	0.010	0.030	0.001	0.096
	Split Dual		**Split Differential**		
	R1	R2	R1	R2	R3
	0.056	0.390	0.067	0.250	0.310

Table 5.1 Varying values of Leak rate (α of equation 5.3) per model. The Split Dual and Split Differential models feature distinct leak rates for each of their ESNs, enabling the establishment of a fast pathway with a larger leak rate and a slow pathway with a smaller leak rate.

performances (top Figure). The values of various leak rates are presented in Table 5.3.1.1, where each reservoir has a distinct leak rate value, highlighting the variability in processing speed. Small leak rates result in a slow update but larger memory, while big leak rates lead to a faster update but smaller memory. However, the performance results indicate that having two distinct parts with different speed processing is not sufficient for achieving an improved working memory. It is crucial not only for the models to possess different leak rates but also for the input to undergo separate processing. Similarly, the Continuous Forward and Continuous Dual models show significantly better performances during trials when the best cue appears first (83% against 76% for the Regular model). This implying that the Split Differential model, Continuous Forward and Continuous Dual have an better working memory than the Regular model.

Performance of no delay task for the bound case (%)						
Regular	Dual	Differential	Split Dual	Split Differential	Continuous Forward	Continuous Dual
93.8	87.8	93.4	86.8	78.0	88.0	88.3

Table 5.2 Model evaluation during the bound no delay scenario. All models were able to perform successfully the task when both stimuli appear simultaneously. The Regular model exhibit the highest performances.



Paired t-test with the Regular model

	Overall	Best first
Differential	0.001	0.008
Dual	0.029	0.353
Split Dual	0.124	**0.01**
Split Differential	**0.007**	**0.0**
Continuous Forward	0.993	**0.002**
Continuous Dual	0.512	**0.001**

Figure 5.7 Model performances for the bound case are compared with the Regular model thanks to a paired t-test. The table show the p values, with the bold values corresponding to the models that exhibit significantly better performances than the Regular one. This includes the Split Differential model overall and when best cue appears first, additionally the Split Dual, the Continuous Forwards and the Continuous Dual only when best cue appears first.

5.3.1.2 Input with 2 by 4 matrix

As shown in Fig. 5.4(right), we can also represent the input by using a 2 by 4 matrix configuration. In this configuration the one 4 length vector is used to represent the cues that are being presented, while the other 4 length vector is used to represent the position. The results from the experiment are detailed in Table 5.3. We see that while the models that use the split input configuration are significantly better than the rest, there are not much significant differences between the models that use the same configuration.

5.4 Discussion

Both the Split Differential Model and Split Dual Model offer significant and borderline significant performance increases respectively over the the Regular networks. Studies have found the presence of interconnected regions in the prefrontal cortex which represent distinct objects in the working memory during memory retrieval tasks, referred to in the literature as stripes. These structures have been used

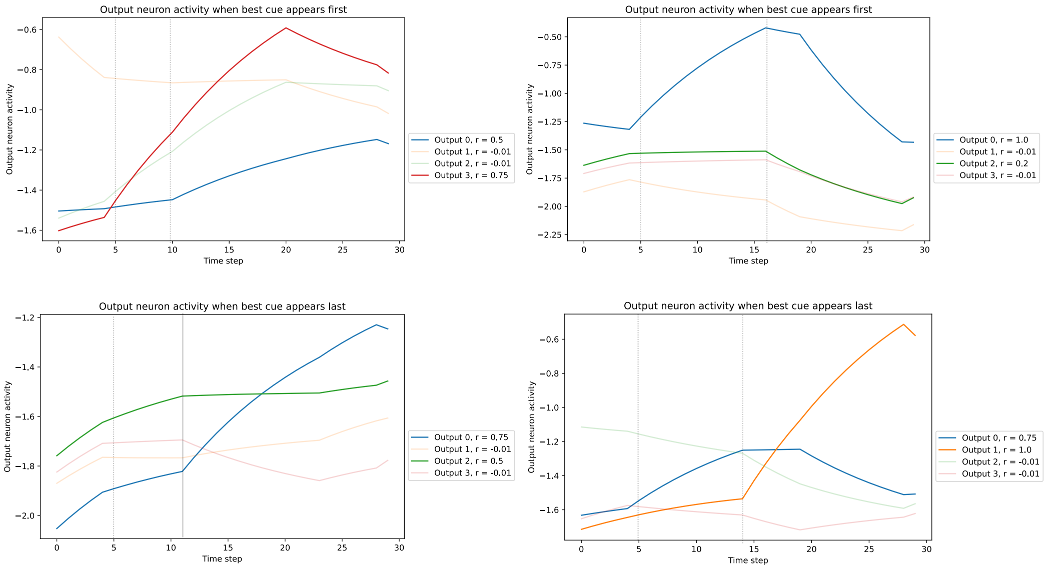


Figure 5.8 Output activity of the Regular model during one trial. The two dotted vertical lines represent the appearance of the first and second cues. The top figures illustrate the scenario when the best cue appears first, while the bottom figure represents when the best cue appears last. A distinct change in activity, indicated by a change in slope, is evident when the second cue appears.

Performance of no delay task for the unbounded case (%)				
Regular	Dual	Differential	Split Dual	Split Differential
43.8	38.5	35.6	81.4	81.8

Table 5.3 Model evaluation during the unbound, no delay scenario. Only the Split Dual and the Split Differential were able to perform successfully the task.Since both stimuli appear simultaneously, the unbound strategy doesn't allow the models to solve the binding problem, except for models that separate the stimuli.

in working memory models [60] as independently updatabale structures which allow the model to represent multiple inputs in its memory at the same time. In the Split reservoir models, used in this work, we see each stripe as a separate reservoir, or in the case of Split Differential, as sets of different reservoirs. The superior performance of the split models showcases the importance of stripe-like structures in temporally challenging tasks such as this.

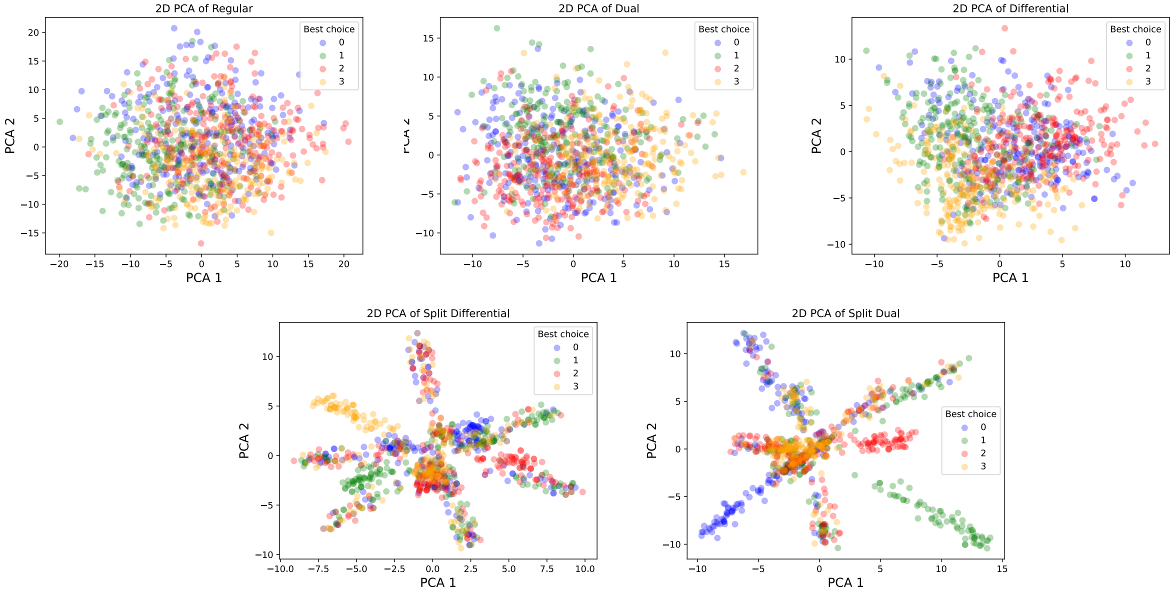
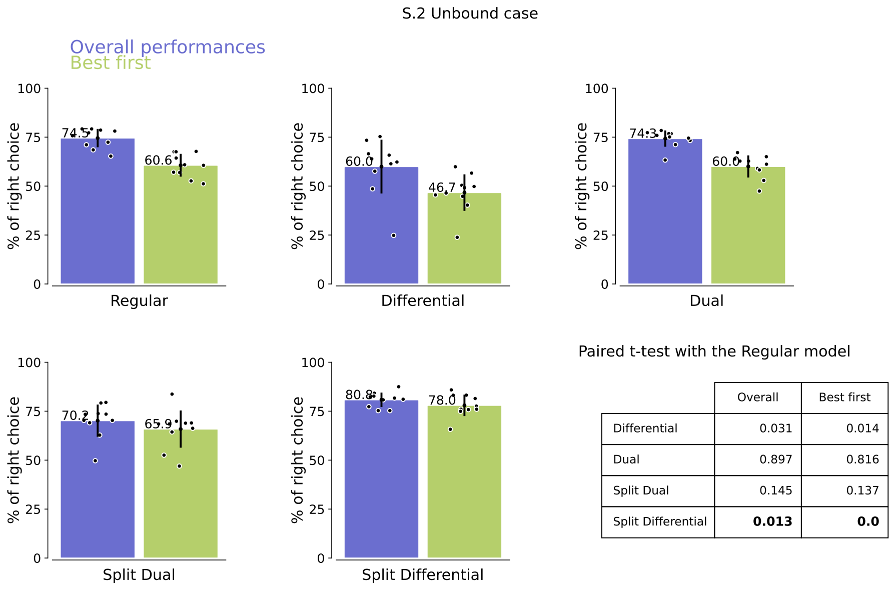


Figure 5.9 2D PCA of individual ESN reservoir states per model. The reservoir states are recorded while the models execute the task during the testing phase, which consists of 1000 trials.

In this task, the decision time is a fixed quantity, in that the model always receives a fixed amount of time steps of input before its output is considered. However, in the real world, animals often take more time for more difficult tasks than easier decisions. While this aspect of decision making has been well studied in cognitive science reflecting on the large amounts of literature available on Evidence Accumulation models, there has also been investigations in the Machine Learning domain in this regard resulting in Anytime Neural Networks. Interestingly, as opposed to Evidence Accumulation Models, where parameters such as threshold are optimised by fitting to be behaviour, anytime neural networks can be trained to appropriately optimise resources so that most compute power is spent on more difficulty input. This is often done through making modifications to the loss function to prioritise latency and these loss functions can be used to train structured reservoirs also. This opens up future investigation of questions like how does the threshold for evidence accumulation change via learning from experiences at a neural level?



Paired t-test with the Regular model

	Overall	Best first
Differential	0.031	0.014
Dual	0.897	0.816
Split Dual	0.145	0.137
Split Differential	**0.013**	**0.0**

Figure 5.10 Model performances for the unbound case are compared with the Regular model via a paired t-test. The table show the p values, with the bold values corresponding to the models that exhibit significantly better performances than the Regular one. This includes the Split Differential model overall and when best cue appears first, additionally the Split Dual, the Continuous Forwards and the Continuous Dual only when best cue appears first.

Chapter 6

Conclusion

Few decisions can be explained by just a single process. Typical decisions we take during daily life often involves processes such as retrieval of memories, and higher-order reasoning. To comprehensively understand why a particular decision was made, one needs to understand all the many processes involved. However, in a wide variety of decision making scenarios, including both perception based and value based decision making, studies have shown ramping brain activations characteristic of Evidence Accumulation processes. Thus, the central idea that motivates this thesis is that decision making across domains happens via an Evidence Accumulation framework. While we are looking to the past or simulating the future in order to solve a problem, we are accumulating information relevant to the task at hand. In the case of perceptual decision making, the mental processes involved can be abstracted as a simple random process whoose drift rate is proportional to the strength of the stimuli. However, as we move towards value based decision making, more sophisticated methods of accumulating needs to be explored.

In this thesis, we primarily focus on how Evidence Accumulation occurs in Value-Based Decision Making particularly in a Reinforcement Learning context. In the first portion of the thesis, the focus is on incorporating Evidence Accumulation into well-established Q-Learning RL models. The proposal is to use the softmaxed Q-values learned by the RL algorithm as the driving force for a sequential sampling process. We find that the combined RL Evidence Accumulation Model is able to perform as well as conventional models in predicting subjects' actions, is able to make good estimates of subjects' reaction times, and is able to produce signiicant differences in goodness-of-fit of various RL models.

In the second portion of the thesis, we look at how Evidence Accumulation models can be realized in Reservoir Neural Networks trained via RL, We investigate several structured reservoirs based on the connectivity patterns seen in the Basal Ganglia. Particularly incorporating the presence of distinct parallel pathways through which activty propagates in a particular direction. We find that such models showcase the characteristic ramping activations. We also discuss the results and its implications for memory in reservoir models.

Overall, the thesis offers preliminary directions for understanding Value-Based decision making via Evidence Accumulation Frameworks which can be expanded upon in the future to more complex tasks and integrated with more sophisticated models of mental processes. RL Evidence Accumulation Models which can utilise reaction time data in addition to choice data offer exciting opportunities for testing RL-related cognitive hypothesis without having to use neuroimaging data or complicated task settings.

Related Publications

- **Reinforcement Learning Linear Ballistic Accumulators as a Model for Grid Navigation**, *Gautham Venugopal*, Bapi Surampudi Raju. *In Annual Conference of Cognitive Science 9 (ACCS) 2022*.
- **Modelling Grid Navigation Using Reinforcement Learning Linear Ballistic Accumulators**, *Gautham Venugopal*, Bapi Surampudi Raju. *In International Joint Conference on Neural Networks (IJCNN) 2023*.
- **Exploring the structure of the Basal Ganglia using Echo State Networks**, Naomi Chaix-Eichel, *Gautham Venugopal*, Boraud Thomas, Nicolas P. Rougier. *In IEEE International Conference on Development and Learning (ICDL) 2024*.

Bibliography

[1] A. Agarwal, A. Kumar, K. Dunovan, E. Peterson, T. Verstynen, and K. Sycara. Evidence Accumulation Allows for Safe Reinforcement Learning, Better Safe than Sorry, 2018.

[2] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.

[3] A. R. Aron and R. A. Poldrack. Cortical and subcortical contributions to stop signal response inhibition: role of the subthalamic nucleus. *Journal of Neuroscience*, 26(9):2424–2433, 2006.

[4] K. Bai, F. Liao, and X. Hu. Reservoir computing with a small-world network for discriminating two sequential stimuli. In *Advances in Neural Networks-ISNN 2017: 14th International Symposium, ISNN 2017, Sapporo, Hakodate, and Muroran, Hokkaido, Japan, June 21–26, 2017, Proceedings, Part I 14*, pages 277–284. Springer, 2017.

[5] I. Bar-Gad, G. Morris, and H. Bergman. Information processing, dimensionality reduction and reinforcement learning in the basal ganglia. *Progress in neurobiology*, 71(6):439–473, 2003.

[6] D. S. Bassett and E. Bullmore. Small-world brain networks. *The neuroscientist*, 12(6):512–523, 2006.

[7] K. Bera, T. Savalia, and B. Raju. *Value-of-Information based Arbitration between Model-based and Model-free Control*. Arxiv, 2019.

[8] K. Bera, A. Shukla, R. S. C. Bapi, and M. Learning. in internally-guided motor skills. *Front*, 12, 2021.

[9] K. A. E. Bera and C. Investigation. *of Skill Learning in Internally-guided Sequencing*. IIIT Hyderabad, 2021.

[10] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

[11] T. Boraud, A. Leblois, and N. P. Rougier. A natural history of skills. *Progress in Neurobiology*, 171:114–124, dec 2018.

[12] S. D. Brown and A. Heathcote. The simplest complete model of choice response time: linear ballistic accumulation. *Cognit. Psychol.*, 57:153–178, 2008.

[13] S. D. Brown and A. Heathcote. The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive Psychology*, 57:153–178, 11 2008.

[14] V. S. Chakravarthy, D. Joseph, and R. S. Bapi. What do the basal ganglia do? a modeling perspective. *Biol Cybern*, 103:237–253, 2010.

[15] Z. Cheng, Z. Deng, X. Hu, B. Zhang, and T. Yang. Efficient reinforcement learning of a reservoir network model of parametric working memory achieved with a cluster population winner-take-all readout mechanism. *Journal of Neurophysiology*, 114(6):3296–3305, 2015.

[16] B. Christian and T. Griffiths. *Algorithms to live by*. Henry Holt & Company, New York, NY, Apr. 2016.

[17] P. Cisek, G. A. Puskas, and S. El-Murr. Decisions in changing conditions: The urgency-gating model. *Journal of Neuroscience*, 29:11560–11571, 2009.

[18] A. G. E. Collins and J. Cockburn. Beyond dichotomies in reinforcement learning. *Nat. Rev. Neurosci.*, 21(10):576–586, Oct. 2020.

[19] M. Dale, S. O'Keefe, A. Sebald, S. Stepney, and M. A. Trefzer. Reservoir computing quality: connectivity and topology. *Natural Computing*, 20:205–216, 2021.

[20] J. D. Davidson and A. El Hady. Foraging as an evidence accumulation process. *PLOS Computational Biology*, 15(7):1–25, 07 2019.

[21] P. Dayan and L. F. Abbott. Theoretical neuroscience (mit press, cambridge. *M*, 2001.

[22] P. G.-d. c. a. Dayan. and its antipodes. *Neural Networks*, 22:213–219, 2009.

[23] Z. Deng and Y. Zhang. Collective behavior of a small-world recurrent neural system with scale-free distribution. *IEEE Transactions on neural networks*, 18(5):1364–1375, 2007.

[24] L. Ding and J. I. Gold. Caudate encodes multiple computations for perceptual decisions. *J Neurosci.*, 30:15747–15759, 2010.

[25] P. F. Dominey, T. M. Ellmore, and J. Ventre-Dominey. Effects of connectivity on narrative temporal processing in structured reservoir computing. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2022.

[26] P. F. Dominey, T. M. Ellmore, and J. Ventre-Dominey. Effects of connectivity on narrative temporal processing in structured reservoir computing. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.

[27] K. Dunovan and T. Verstynen. *Believer-Skeptic meets Actor-Critic: Rethinking the role basal ganglia pathways in decision-making and reinforcement learning*, 10:1101, 2016.

[28] N. J. Evans and E.-J. Wagenmakers. Evidence accumulation models: Current limitations and future directions. *Quant. Methods Psychol.*, 16(2):73–90, Apr. 2020.

[29] A. Fermin, T. Yoshida, M. Ito, J. Yoshimoto, and K. Doya. Evidence for model-based action planning in a sequential finger movement task. *J Mot BehavNov;*, 42(6):371–9, 2010.

[30] A. Fermin, T. Yoshida, M. Ito, J. Yoshimoto, and K. Doya. Evidence for model-based action planning in a sequential finger movement task. *Journal of Motor Behavior*, 42:371–379, 2010.

[31] A. S. R. Fermin, T. Yoshida, J. Yoshimoto, M. Ito, S. C. Tanaka, and K. Doya. Model-based action planning involves cortico-cerebellar and basal ganglia networks. *Sci. Rep.*, 6(1), Aug. 2016.

[32] L. Fontanesi, S. Gluth, M. S. Spektor, et al. A reinforcement learning diffusion decision model for value-based decisions. *Psychon. Bull*, 26:1099–1121, 2019.

[33] M. J. Frank, J. Samanta, A. A. Moustafa, and S. J. Sherman. Hold your horses: Impulsivity, deep brain stimulation, and medication in parkinsonism. *Science*, 318:1309–1312, 2007.

[34] C. Gallicchio, A. Micheli, and L. Pedrelli. Deep reservoir computing: A critical experimental analysis. *Neurocomputing*, 268:87–99, 2017.

[35] S. J. Gershman. Do learning rates adapt to the distribution of rewards? *Psychon Bull Rev*, 22:1320–1327, 2015.

[36] J. Gl"ascher, N. Daw, P. Dayan, and O. JP. States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron*, 66:585–595, 2010.

[37] J. I. Gold and M. N. Shadlen. The neural basis of decision making. *Annu. Rev. Neurosci.*, 30(1):535–574, 2007.

[38] A. Heathcote, S. J. Popiel, and D. J. Mewhort. Analysis of response time distributions: An example using the stroop task. *Psychological Bulletin*, 109:340–347, 1991.

[39] P. C. Holland. Relations between pavlovian-instrumental transfer and reinforcer devaluation. *J Exp Psychol Anim Behav Process*, 30:104–117, 2004.

[40] H. Jaeger. Echo state network. *scholarpedia*, 2(9):2330, 2007.

[41] H. Jaeger. Echo state network. *scholarpedia*, 2(9):2330, 2007.

[42] S. Jarvis, S. Rotter, and U. Egert. Extending stability through hierarchical clusters in echo state networks. *Frontiers in neuroinformatics*, 4:11, 2010.

[43] Y. Kawai, J. Park, and M. Asada. A small-world topology enhances the echo state property and signal propagation in reservoir computing. *Neural Networks*, 112:15–23, 2019.

[44] H. Kim, J. H. Sul, N. Huh, D. Lee, and M. W. Jung. Role of striatum in updating values of chosen actions. *Journal of neuroscience*, 29:47, 2009.

[45] K.-i. Kitayama. Guiding principle of reservoir computing based on “small-world” network. *Scientific reports*, 12(1):16697, 2022.

[46] W. Kool, F. A. Cushman, and S. J. Gershman. Competition and cooperation between multiple reinforcement learning systems. *Goal-Directed Decision Making: Computations and Neural Circuits*, pages 153–178, 2018.

[47] I. Krajbich, B. Bartling, T. Hare, and E. Fehr. Rethinking fast and slow based on a critique of reaction-time reverse inference. *Nature communications*, 6:1, 2015.

[48] J. Lampinen. A constraint handling approach for the differential evolution algorithm. In *Proceedings of the 2002 Congress on Evolutionary Computation*. CEC’02 (Cat. No. 02TH8600). Vol. 2. IEEE, 2002.

[49] B. Liebald. Exploration of effects of different network topologies on the esn signal crosscorrelation matrix spectrum. *Bachelor of Science (B. Sc.) thesis, International University Bremen, spring*, 2004.

[50] M. Lukoševičius. A practical guide to applying echo state networks. In *Neural Networks: Tricks of the Trade: Second Edition*, pages 659–686. Springer, 2012.

[51] R. B. Mars, N. J. Shea, N. Kolling, and R. MFS. Model-based analyses: Promises, pitfalls, and example applications to the study of cognitive control. *Q. J. Exp*, 65:252–267, 2012.

[52] J. L. McClelland and D. E. Rumelhart. An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychol. Rev.*, 88(5):375–407, Sept. 1981.

[53] S. Miletić, R. J. Boag, and B. U. Forstmann. Mutual benefits: Combining reinforcement learning with sequential sampling models. *Neuropsychologia*, 136(107261):107261, Jan. 2020.

[54] P. J. Mineault, S. Bakhtiari, B. A. Richards, and C. C. Pack. Your head is there to move you around: Goal-driven models of the primate dorsal pathway. July 2021.

[55] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, Feb. 2015.

[56] J. Moon, Y. Wu, and W. D. Lu. Hierarchical architectures in reservoir computing systems. *Neuromorphic Computing and Engineering*, 1(1):014006, 2021.

[57] M. J. Nissen and P. Bullemer. Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology*, 19(1):1–32, 1987.

[58] Y. Niv. Reinforcement learning in the brain. *J. Math. Psychol.*, 53(3):139–154, June 2009.

[59] M. Oaksford and N. Chater. *Bayesian Rationality*. Oxford Cognitive Science Series. Oxford University Press, London, England, Mar. 2007.

[60] R. C. O'Reilly and M. J. Frank. Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia. *Neural Comput.*, 18(2):283–328, Feb. 2006.

[61] S. Pabst, M. Brand, and O. T. Wolf. Stress and decision making: A few minutes make all the difference. *Behavioural brain research*, 250:39–45, 2013.

[62] I. P. Pavlov. Excerpts from the work of the digestive glands. *Am. Psychol.*, 52(9):936–940, Sept. 1997.

[63] M. L. Pedersen, M. J. Frank, and G. Biele. The drift diffusion model as the choice rule in reinforcement learning. *Psychonomic Bulletin & Review*, 24:1234–1251, 2017.

[64] R. L. Perri, M. Berchicci, D. Spinelli, and F. Di Russo. Individual differences in response speed and accuracy are associated to specific brain activities of two interacting systems. *Front*, 8, 2014.

[65] M. A. Pisauro, E. Fouragnan, C. Retzler, and M. G. Philiastides. Neural correlates of evidence accumulation during value-based decisions revealed via simultaneous eeg-fmri. *Nature Communications*, 8:15808, 6 2017.

[66] R. Ratcliff and G. McKoon. The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, 20:873–922, 4 2008.

[67] R. Ratcliff and G. McKoon. The diffusion decision model: theory and data for two-choice decision tasks. *Neural computation*, 20, 2008.

[68] R. Ratcliff and J. N. Rouder. Modeling response times for two-choice decisions. *Psychological Science*, 9:347–356, 9 1998.

[69] R. Ratcliff, A. Thapar, P. Gomez, and G. A. McKoon. diffusion model analysis of the effects of aging in the lexical-decision task. *Psychol. Aging*, 19:278–89, 2004.

[70] A. Ratcliff R.:. theory of memory retrieval. *Psychological Review*, 85:59–108, 1978.

[71] P. Redgrave. Basal ganglia. *Scholarpedia J.*, 2(6):1825, 2007.

[72] R. Rescorla and A. Wagner. *A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement*, volume Vol. 2. 01 1972.

[73] N. Rodriguez, E. Izquierdo, and Y.-Y. Ahn. Optimal modularity and memory capacity of neural reservoirs. *Network Neuroscience*, 3(2):551–566, 2019.

[74] N. P. Rougier. A density-driven method for the placement of biological cells over two-dimensional manifolds. *Frontiers in Neuroinformatics*, 12, 3 2018.

[75] A. Roxin. Drift–diffusion models for multiple-alternative forced-choice decision making. *J. Math. Neurosc.*, 9, 2019.

[76] R. Schmidt, D. K. Leventhal, N. Mallet, F. Chen, and J. D. Berke. Canceling actions involves a race between basal ganglia pathways. *Nature neuroscience*, 16(8):1118–1124, 2013.

[77] N. W. Schuck, R. Gaschler, D. Wenke, J. Heinzle, P. A. Frensch, J.-D. Haynes, and C. Reverberi. Medial prefrontal cortex predicts internally driven strategy shifts. *Neuron*, 86(1):331–340, Apr. 2015.

[78] W. Schultz, P. Apicella, and T. Ljungberg. Responses of monkey dopamine neurons to reward and conditioned stimuli during successive steps of learning a delayed response task. *J. Neurosci.*, 13(3):900–913, Mar. 1993.

[79] W. Schultz, P. Dayan, and P. R. A. Montague. neural substrate of prediction and reward. *Science*, 275:1593–1599, 1997.

[80] G. Schwarz. Estimating the dimension of a model. *Ann*, 6:461–464, March 1978.

[81] M. N. Shadlen and W. T. Newsome. Neural basis of a perceptual decision in the parietal cortex (area lip) of the rhesus monkey. *Journal of neurophysiology*, 86:4, 2001.

[82] D. Sheynikhovich. Spatial navigation in geometric mazes: a computational model of rodent behavior. 01 2007.

[83] R. S. Sutton and A. G. R. L. A. I. Barto. MIT Press, The, 2018.

[84] S. Teerapittayanon, B. McDanel, and H. T. Kung. Branchynet: Fast inference via early exiting from deep neural networks. *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2464–2469, 2016.

[85] M. Theisen, V. Lerche, M. von Krause, et al. Age differences in diffusion model parameters: a meta-analysis. *Psychological Research*, 85:2012–2021, 2021.

[86] E. L. A. I. E. s. M. Thorndike. 1911.

[87] D. Thura, J. Beauregard-Racine, C. W. Fradet, and P. Cisek. Decision making by urgency gating: Theory and experimental support. *Journal of Neurophysiology*, 108:2912–2930, 2012.

[88] M. Topalidou, D. Kase, T. Boraud, and N. P. Rougier. A computational model of dual competition between the basal ganglia and the cortex. *eneuro*, 5(6), 2018.

[89] E. Tricomi, B. W. Balleine, and J. P. A. O'Doherty. specific role for posterior dorsolateral striatum in human habit learning. *Eur*, 29:2225–2232, 2009.

[90] N. Trouvain, L. Pedrelli, T. T. Dinh, and X. Hinaut. Reservoirpy: an efficient and user-friendly library to design echo state networks. In *International Conference on Artificial Neural Networks*, pages 494–505. Springer, 2020.

[91] M. Usher and J. L. McClelland. The time course of perceptual choice: The leaky, competing accumulator model, 2001.

[92] M. Usher and J. L. McClelland. The time course of perceptual choice: the leaky, competing accumulator model. *Psychol. Rev.*, 108:550–92, 2001.

[93] L. van Maanen, S. D. Brown, T. Eichele, E.-J. Wagenmakers, T. Ho, J. Serences, and B. U. Forstmann. Neural correlates of trial-to-trial fluctuations in response caution. *Journal of Neuroscience*, 31:48, 2011.

[94] D. Vickers. Evidence for an accumulator model of psychophysical discrimination. *Ergonomics*, 13(1):37–58, 1970.

[95] A. Wald and J. Wolfowitz. Optimum Character of the Sequential Probability Ratio Test. *The Annals of Mathematical Statistics*, 19(3):326 – 339, 1948.

[96] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[97] S. Weissengruber, S. W. Lee, J. P. O'Doherty, and C. C. Ruff. Neurostimulation reveals context-dependent arbitration between model-based and model-free reinforcement learning. *Cerebral Cortex*, 29(11):4850–4862, 2019.

[98] A. Wiehler and P. Diffusion. *modeling reveals reinforcement learning impairments in gambling disorder that are linked to attenuated ventromedial prefrontal cortex value representations. biorxivv*. Jan, 2020.

[99] Wikipedia. Anytime algorithm — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Anytime%20algorithm&oldid=1166967923`, 2023. [Online; accessed 30-December-2023].

[100] R. A. Wise, J. Spindler, H. deWit, and G. J. Gerberg. Neuroleptic-induced "anhedonia" in rats: pimozide blocks reward quality of food. *Science*, 201(4352):262–264, July 1978.

[101] K. Wunderlich, P. Dayan, and R. J. Dolan. Mapping value based planning and extensively trained choice in the human brain. *Nat. Neurosci.*, 15:786–791, 2012.

[102] K. Wunderlich, A. Rangel, and J. P. O'Doherty. Neural computations underlying action-based decision making in the human brain. *Proceedings of the National Academy of Sciences*, 106(40):17199–17204, 2009.

[103] Y. Xue, L. Yang, and S. Haykin. Decoupled echo state networks with lateral inhibition. *Neural Networks*, 20(3):365–376, 2007.

[104] S. Zhang, M. D. Lee, W. E.-J. Vandekerckhove, J., and G. Maris. Time-varying boundaries for diffusion models of decision making and response time. *Frontiers in Psychology*, 5, 2014.