Concepts in Bounded Rationality: Perspectives from Reinforcement Learning

David Abel

Master's Thesis

Submitted in partial fulfillment of the requirements for the

Degree of Master of Arts in the Department of Philosophy at Brown University

Providence, Rhode Island

May 2019

This thesis by David Abel is accepted in its present form by the Department of Philosophy as satisfying the thesis requirements for the degree of Master of Arts.

Date ________________ ____________________________________
Joshua Schechter, Advisor

Approved by the Graduate Council

Date ________________ ____________________________________
Andrew G. Campbell, Dean of the Graduate School

Acknowledgements

I am beyond thankful to the many wonderful mentors, friends, colleagues, and family that have supported me along this journey:

- A profound thank you to my philosophy advisor, Joshua Schechter – his time, insights, patience, thoughtful questioning, and encouragement have been essential to my own growth and the work reflected in this document. He has been a brilliant mentor to me and I am grateful for the opportunity to learn from him.

- I would like to thank my previous philosophy advisors: from highschool, Terry Hansen, John Holloran, and Art Ward, who first gave me the opportunity to explore philosophy and ask big questions; and to my undergraduate mentors, Jason Decker, David Liben-Nowell, and Anna Moltchanova, all of whom helped me grow as a thinker and writer and encouraged me to pursue grad school.

- A deep thank you to my recent collaborators, mentors, and colleagues who have contributed greatly to the ideas presented in this work: Cam Allen, Dilip Arumugam, Kavosh Asadi, Owain Evans, Tom Griffiths, Ellis Hershkowitz, Mark Ho, Yuu Jinnai, George Konidaris, Michael Littman, James MacGlashan, Daniel Reichman, Stefanie Tellex, Nate Umbanhowar, and Lawson Wong. A special thanks to Mark Ho for the many thoughtful conversations involving the topics explored in this thesis, and to my CS Ph.D advisor Michael Littman for giving me the freedom and support to pursue a Masters in philosophy during my PhD.

- I would like to thank Richard Heck for first encouraging me to pursue a Masters in philosophy, and for being a fantastic teacher; I would not have decided to go through the program without his support.

- Thanks to all of other my previous philosophy teachers: David Christensen, Jamie Dreier, Roy Elveton, Daniel Groll, Chris Hill, Sarah R. Jansen, Colin G. King, and Adam Pautz. An extra thank you to Chris Hill for helping me navigate the philosophy program in his role as DGS.

- Thanks to Alejandro Prez Carballo for insightful discussions that helped shape aspects of this work.

- Lastly, I extend my deepest gratitude to my mom (Diane), dad (Mark), brother (Michael), girlfriend (Elizabeth), and close friends; their support, insights, curiosity, and kindness have been essential to every part of this work, and to my love of learning.

Thank you, everyone!

Contents

List of Figures

Abstract of "Concepts in Bounded Rationality: Perspectives from Reinforcement Learning", by David Abel, A.M., Brown University, May 2019.

In this thesis, I explore the relevance of computational reinforcement learning to the philosophy of rationality and concept formation. I first argue that the framework of reinforcement learning can be used to formalize an intuitive notion of bounded rationality, thereby enabling new analysis of rational agency under realistic assumptions. I then show how the process of concept formation is intimately connected to an agent's capacity to be rational, subject to relevant resource constraints. Together, these claims suggest that rational agents must adopt a reasonable set of concepts. The main contribution of this work is an initial characterization of what constitutes a good set of concepts for a resource constrained rational agent: namely, the set makes the appropriate trade-off between simplicity (so planning and learning are easy) and representational power (so decisions are effective).

Chapter 1

Introduction

Understanding the nature of good reasoning is a fundamental goal of philosophy, along with major subfields of computer science, economics, and beyond. Often, these studies concentrate on highly *idealized* models of rational behavior or belief: what would the perfect agent do in the same situation? Such idealizations can serve as a guide for clarifying our own belief forming and decision making practices [28].

However, there is more to reasoning than just this golden standard. A holistic theory of rationality is likely to benefit from sharpened assumptions that better capture the situation most agents face, such as constraining the amount of computation available or limiting prior world knowledge. To this end, alternate theories have sought more realistic characterizations of good reasoning. The bounded rationality framework introduced by Simon [138] and its kin, such as bounded optimality [129], minimal rationality [27], ecological rationality [51, 50], and computational rationality [49] all provide strong arguments for focusing attention on something other than idealized rationality. As stated by Christopher Cherniak, the proponent of minimal rationality: "The assumed conception of rationality in philosophy is too idealized to be applied to humans" (pp. 163). I take this critique to be correct: with more realistic assumptions, we can draw even more relevant insights to our own lives and communities. Further, psychological evidence supports the thesis that people regularly act in a way that departs from what idealized models suggest, making heavy use of inductive biases and heuristics to make fast, useful, but occasional non-ideal decisions [78, 167, 77].

Through the appropriate formalism, we can analyze boundedly rational behavior and belief updating much in the same way that logic, probability theory, and decision theory have given profound structure and insight to our broader understanding of good belief formation and decision making practices. Each of these alternative theories of rationality proposes adjustments that more acutely describe rational behavior. I will discuss some of their nuances in Chapter 2.

I will then turn to recent advances in artificial intelligence research, which have shed light on mechanistic approaches to learning, belief, and problem solving, subject to realistic constraints on reasoning. These insights have given rise to algorithms that have empowered our understanding belief formation and decision making under realistic assumptions; these include the field of reinforcement learning (RL), a general problem formation in which agents must simultaneously learn about their environment while making good decisions in that environment [75, 154, 155].

The first goal of this thesis is to illustrate the power of computational reinforcement learning for illuminating the nature of reasoning under realistic assumptions. The second goal is to shed light on the role concepts play in bounded rationality, based on insights from reinforcement learning. In particular, I advance the following thesis:

Thesis: Bounded rationality is not solely about elegant symbol crunching, but also involves choosing the right concepts so that symbol crunching is *easy* and *effective*. Moreover, we can understand "right concepts" precisely in terms of the extent to which concepts make the calculations relevant to decision making both easy and effective.

To defend this thesis, I argue for three claims. First, I motivate reinforcement learning (RL) [154, 75, 155], as a suitable formalism for studying boundedly rational agents. Second, in light of this formalism, I argue that the formation of appropriate concepts is an essential aspect of any rational resource-bounded agent. Third, I offer an initial theory about what constitutes a set of *good concepts* for boundedly rational agents based on trading off for representative power and simplicity.

In more detail, I first argue that RL can serve as a theoretical tool for analyzing bounded rational agents. At a high level, RL unifies learning and decision making into a single, general framework. Specifically, RL asks how agents can simultaneously learn about their environment while making good decisions in that environment. In RL, agents are often subject to constraints on processing power, time, and memory, and are only given limited initial knowledge about the world, the world's ontology, and the world's causal laws. In this sense, RL serves as a generalization of typical theories of decision making [121]. How does an agent come to learn the consequences of its actions in an environment that constantly changes? How can an agent form general concepts that will enable quick and accurate predictions? Through such a general learning framework, I take advantage of known results in computational complexity theory [7] and computational learning theory [168, 81], which provide mathematical insight into the nature of accurate belief formation and problem solving subject to constraints in the real world [1, 60]. RL adopts the formal tools from both fields, which gives it a privileged vantage to offer a unifying theory of how agents come to understand their worlds well enough to solve problems in them. This is precisely the connection I will exploit in order to formalize bounded rationality.

Second, as a consequence of taking RL as a model of bounded rationality, I argue that good concept formation is a necessary condition for being a rational agent; I claim a concept is good just when it empowers agents to make better decisions subject to the same resource constraints, building in part on the insights of Relevance Theory, developed by Wilson and Sperber [174]. I here take "concept" to be the psychological entities that an agent can entertain through thoughts, though I will later divide on two broad categories: 1) concepts about *world state*, and 2) concepts about *agent behavior*, which lets us better align with the tools of RL. I build on existing inquiries into concept formation that ask: "what constitutes the right concepts rational agents should form?" [21]. I argue that being a bounded rational agent necessitates the formation of good concepts, building on recent work in RL [59]. For example, suppose an agent has access to only a small amount of memory and must plan a route to work in a busy city. Coming up with a good plan of action involves reasoning about a combination of activities, including walking, navigating through buildings, and using public transportation. For different choices of how to break down the constituents of this plan (buildings, doors, crowds, lights, and so on), the agent will speculate over different simulated routes: if the agent reasons over only their fine-grained motor controls, then considering the forward-search-tree of all possible paths into the future will quickly exhaust all of the agent's available memory (for even generous memory constraints). However, if an agent instead forms concepts that make an appropriate balance between explaining relevant phenomena accurately and succinctly, the agent will be able to search over and find a high quality sequence of actions *efficiently.* To do so requires appropriate concepts.

Third, I close by offering an initial theory of how to characterize good concepts through RL. The core of the theory stipulates that agents should seek out concepts that are: 1) simple, so as to make decision making efficient, but 2) effective, so that the results of making decisions using such concepts are good. I draw on the tools of information theory to use *compression* as a means of making this trade-off, building on the work of Ortega Jr [117], Ortega and Braun [116] who present a unified account of information theory and bounded rationality. Our problem is to jointly identify representations for different *states* of the world and to model an agent's own *behavior* in an appropriate way. In the example of an agent navigating to work in a busy city, the agent should find a set of concepts that characterize world states and the agent's own behaviors that are needed in order to accurately distinguish between plans that take the agent to work, and plans that take the agent elsewhere. Critically, such concepts must be parsimonious so as to minimize the resulting computational costs of operating in the space of those concepts.

To summarize, this work is about defending three claims:

1. Bounded rationality can be adequately characterized in terms of RL.

2. If an agent with finite resources (time, memory, perceptual bandwidth) is considered rational, then by necessity the agent has formed good concepts.

3. RL suggests a desirable theory of boundedly rational concepts: a concept is rational just when it supports trade-off between *compression* and *representational power*.

The rest of this thesis is as follows. I first (Chapter 2) provide necessary background on ideal rationality. I also summarize bounded rationality and its neighboring theories–minimal rationality, and ecological rationality. I also give background on computational complexity, computational learning theory, and RL, which will serve as the core formalisms for much of the work. The second chapter of the paper is dedicated to introducing, motivating, and defending RL as a reasonable mathematical language for studying rationality (Chapter 3), in addition to describing some consequences of taking RL as our central model of agency. In Chapter 4, I explore how the practice of forming concepts [21] is a fundamental project of any would-be realistic rational agent. I suggest that RL gives us a foothold into evaluating theories of concept formation. In Chapter 5, I close by introducing a partially realized theory for good concepts.

Chapter 2

Background

The ability to reason carefully about a diversity of affairs is a hallmark of generally intelligent behavior. Indeed, it is an essential characteristic of our being that differentiates us from animals. As Christensen states in the opening of his book, *Putting Logic In Its Place*, "If there is one respect in which humans differ most fundamentally from the other animals, perhaps it is our superior ability to reason about, and understand, our world" [28].

The study of our "ability to reason about, and understand, our world" has received great again from a variety of disciplines, including epistemology, neuroscience, economics, biology, and machine learning. In this work, we subdivide our study of rationality into that of *practical* rationality, which focuses on making rational *decisions* [109, 57], and *epistemic* rationality, which focuses on aligning one's *beliefs* in a rational way [28]. My study here is primarily concerned with practical rationality. In practical rationality, we focus on how an agent chooses its action, a, given (partial) knowledge of the current state of the world s and some set of competing alternatives, $\mathcal{A}$. In epistemic rationality, we focus on the sets of beliefs are most *rational* given a body of evidence, or alternatively, evaluate whether a group of beliefs B are *more* or *less* rational than another group, C.

What might it mean for beliefs or actions to be rational? A typical strategy involves describing conditions for rational belief formation or action selection from the perspective of the ideally rational agent, giving rise to theories of *ideal rationality*.

2.1 Ideal Rationality

The nature of ideal rationality differs between epistemic rationality and practical rationality. We here present both canonical ideal agents–as we will see, however, both variants make similar background assumptions that limit the usefulness of their respective theory.

2.1.1 Ideal Epistemic Rationality

In epistemic rationality, the ideal rational agent is said to always perfectly align its beliefs to the available bodies of evidence. "Perfect alignment", naturally, can have several interpretations.

One standard definition for perfect alignment is that of adhering to *probabilistic coherence*; any rational agent's set of beliefs must always satisfy the axioms of probability:

1. For every proposition X, $0 \leq \Pr(X) \leq 1$.

2. If X is a tautology, then $\Pr(X) = 1.0$.

3. If X and Y are independent, then $\Pr(X \cup Y) = \Pr(X) + \Pr(Y)$.

Here, the probabilities denote an agent's *credences*, indicating the degree to which the agent believes the proposition X to be true.

The above criteria offer a natural test for determining whether an agent is rational: does the agent's beliefs satisfy the above conditions? If not, then the agent is not rational. Critically, it might be that the prescribed properties are *necessary* for rationality, but not sufficient.

More generally, as Christensen states, rational beliefs are those that result from *good* thinking: “Rational beliefs, it seems, are those arising from good thinking, whether or not that thinking was successful in latching on to the truth” [28]. But, as Christensen goes on to note, this effectively kicks our can down the road: what underlies good thinking? What differentiates it from bad? Christensen gives an initial answer (which he later disagrees with) that we might suppose good thinking is *logical* thinking. If an agent is following an appropriate choice of logical rules, the agent is said to be undertaking *good* thinking. Here, “good thinking” might mean that the agent is logically consistent, or that their beliefs are closed under deduction. Hence, no rational agent can believe in a contradiction, and must also correctly believe that all true claims of a given formal system are true. Some version in the neighborhood of these properties underlies most pictures of ideal epistemic rationality—we find a set of necessary conditions that must hold of the agent’s beliefs for an arbitrary body of evidence.

Suppose you are presented with an immensely complex logical statement that happens to be a tautology. It is reasonable to assume that the complexity inherent to the statement prevents us (and any rational agent) from immediately believing its truth. Without properly working out a proof, it is perhaps *irrational* to believe such a statement, until you are adequately convinced!

Consider the following rather extreme example:

Example 1. *Suppose you come up with a new symbolic system consisting of the alphabet,* $\{a, b, c\}$, *the axiom, aac, and the following rules:*

$$(1)\ aa \implies bb \tag{2.1}$$

$$(2)\ ab \implies c \tag{2.2}$$

$$(3)\ c \implies ba \tag{2.3}$$

$$(4)\ b \implies ca \tag{2.4}$$

$$(5)\ a \implies ccc \tag{2.5}$$

You are asked: are either of the following theorems of the above system?

$$(i)\ abcabc \tag{2.6}$$

$$(ii)\ aabc \tag{2.7}$$

Under the criteria of probabilistic coherence or deductive closure, any belief set *other* than the one that exactly matches the Truth is said to be irrational. The difficulty of this example is that, surely, *any* agent must spend some amount of time determining whether each of (i) and (ii) are theorems. Surely you can't be held accountable for having to work through the rules to identify a satisficing path that leads from the axiom to either (i) or (ii). Indeed, reaching a conclusion without working out any of the relevant details seems epistemically irresponsible. But, how can any agent hope to achieve such a feat, if there are arbitrarily many consequences of even basic axiomatic systems?

Such is the problem of logical omniscience [147, 67, 45]. Should an agent be expected to immediately know all of the consequences of a given axiomatic system? Some have answered this question in the affirmative, such as Smithies [147], arguing that *a priori* justification in favor of tautologies is strong enough to overcome concerns about omniscience (*a priori* true claims can be justified in virtue of their *a priori* truth!). We will return to this issue in Section 2.1.3.

Lastly, it is important to distinguish between diachronic and synchronic explanations of belief, and their role in rationality. Diachronic rationality refers to the (rational) process of *updating* one's beliefs. Suppose an agent's current beliefs are based on some evidence collected over time, $E_1, \ldots, E_t$. Then, given a new body of evidence, E_{t+1}, how should the agent *update* their beliefs in light of the new evidence? A typical response might be to invoke Bayesian Conditionalization as an appropriate belief updating rule (see Chapter 4 of Titelbaum [164] for more details). Conversely, synchronic rationality refers to the process of rationally maintaining one's belief at a particular moment in time (see Chapters 1-3 of Titelbaum [164] for more background). That is, ensuring that your beliefs are contradiction free, for instance. In both variants of belief, the problems with the ideal persist in roughly the same form.

Our first take on ideal rationality prescribes some set of necessary conditions on rationality as some appropriate set of criteria to meet. As in the problem of logical omniscience, idealized rationality tends to bestow agents with unlimited resources, which is problematic. Fixing this problem is the primary aim of bounded rationality.

2.1.2 Ideal Practical Rationality

Our other variant of ideal rationality concentrates on making good decisions, and so finds its roots in economics, rational choice theory, and decision theory [40, 172, 17]. We now suppose an agent is tasked with making the right decision given a set of competing alternatives. Again, we need to clarify what "right decision" means. The usual formalism for such a problem assigns a utility function to different outcomes; then, a decision making agent is tasked with making choices over time so as to maximize its utility. We imagine there exists a set of possible choices, $\mathcal{A} = \{a_1, \ldots, a_n\}$, and a set of possible world states, $\mathcal{S}$. Then, the agent must make a choice $a \in \mathcal{A}$, and is evaluated based on the expected utility of its choice. The optimal choice is then defined as the action that maximizes expected utility:

$$a^* := \max_{a \in \mathcal{A}} \sum_{s \in} U(s) \Pr(s \mid a), \tag{2.8}$$

for $U : \mathcal{S} \to \mathbb{R}$ a utility function, expressing the desirability of each given state of affairs, $s \in \mathcal{S}$, and $\Pr(s \mid a)$ denotes the probability of action a leading to world state s.

In this simple formulation of utility maximization, we can compare the utility of the agent's choice $\hat{a}$ to the optimal choice a^*:

$$U(\hat{a}) - U(a^*), \tag{2.9}$$

which yields a direct measurement of the desirability of an agent's choice. For more on decision theory and its variants, see Steele and Stefnsson [149].

Critically, if the agent is uncertain about either $U(s)$ or $\Pr(s \mid a)$, things become more difficult. This poses the problem of decision making under uncertainty, as studied by Knight [82]. If we further generalize Knight's setting to include an arbitrarily long sequence of consequences, we find ourselves in the setting of *sequential decision making under uncertainty*. That is, when an agent takes an action, we model how the world might change as a result (and in turn, present the agent with a new decision problem, one step forward in time). Here, agents need to consider not just the immediate consequences of their actions but also the long term consequences. When the agent does not know how the world works (the causal rules), this decision making paradigm is popularly formalized by computational reinforcement learning. We will attend to the full details of this subject in Section 2.5.

However, the same problem of logical omniscience rears its head again: how is an agent to know, *a priori*, the relevant quantities needed to find the best action? We again assume that the ideal (practical) rational agent knows the utility of every action it might execute in every state, and acts according to a^* at all times. This is perhaps a useful construct to have defined, but is not helpful for illustrating rational behavior to any agent with resource constraints; suppose, for instance, that an agent can only hold a certain number of action utilities in its head at once. In this case, how are we to act? To answer this question we will next turn to the bounded rationality framework.

To summarize, these two characterizations broadly represent idealized theories of rationality: either an agent is assessed according to its capacity to make decisions with respect to some utility function (pragmatic rationality), or is evaluated according to the extent to which its belief forming/updating methods meet a certain set of criteria, such as closure under deduction or Bayesian conditionalization (epistemic rationality). Naturally, the ideal in both cases is said to outperform all other approaches – the ideal pragmatic agent always chooses $U(a^*)$, and the ideal epistemic agent always maintains correct beliefs. As highlighted, these ideals are not without issue.

2.1.3 The Purpose of Ideal Rationality

It is worth briefly discussing the purpose of studying rationality. Typically, rationality is used to clarify what *could*, in principle, be done in response to the core practices of the mind, with a focus on deduction, belief forming, and decision making. Theories of rationality are often out to achieve multiple different objectives: in some cases, we might care about clarifying what is the normative "correct" thing to do in response to some stimuli. This notion of "correct" might be used to guide people in their own reasoning and belief formation practices. For example, if you hear thunder, it might be appropriate to believe that the weather outside will take a particular form. It would be irrational to suppose that someone were simulating thunder sounds without other evidence to make such a situation more feasible.

To better highlight the issues at play, let us consider the different roles rationality can play.

First, rationality can be used as a guide for our reasoning. We hope to bolster our ability to seek out appropriate knowledge, and to make good decisions in the face of uncertainty. In what sense does ideal rationality provide is with meaningful instruction as to how to update our beliefs or make good decisions? We lack the ability to introspect and identify the precise real numbers corresponding to our beliefs, but more importantly, Bayesian belief updating is known to be intractable in even simple domains. So, again, we find ourselves failing to find usefulness in practice. Properties like deductive closure or probabilistic coherence give rise to issues concerning logical omniscience for nearly any symbolic system of relevance. But of course there is some use in these ideals: we can clarify what should be done, in principle, which can give us a clear theoretical picture of what, under realistic assumptions, we can hope to achieve. While we can't expect to do full on Bayesian belief updating for all evidence, we can assert that we should be *approximating* something *like* these kinds of methods. It is in this sense that these ideals are highly informative: they can make concrete our objectives when we operate under realistic assumptions.

Second, rationality can be used to place blame on others when they deviate wildly from certain norms. This motivation raises more questions than it answers. Consider logical omniscience: surely no person is responsible for assigning knowing all True propositions of a given system. But if we can't hold them blameworthy for this, what *can* we? Is there not always an out, that we're not ideal reasoners so mistakes are inevitable? Surely there is hope here. In the same way that ideal rationality helps clarify our objectives of rationality under realistic assumptions, so to can we find how to place blame for failing to be rational under realistic assumptions: if an agent *has* the relevant computing time, requisite knowledge, and acumen, they can be held blameworthy for making highly sub-optimal decisions.

Third, it can be useful to clarify what the perfect form of rationality looks like to understand what improvements we should carry out. Often we are trying to move closer to some golden standard, which variations of rationality can give us.

To conclude, ideal rationality is still useful insofar as it informs the objectives of rationality under realistic assumptions. We can then form and evaluate appropriate approximations according to the sense in magnitude that they deviate from the ideal under the relevant assumptions. We next introduce some of these previous theories.

2.2 Alternative Theories

Many alternative theories have been proposed, each of which attempting to remedy different shortcomings of idealized rationality by studying rational agents in a more realistic setting. The origin of this line of study comes from bounded rationality, first introduced by Simon [138], which proposed to characterize an agent's capacity for rational behavior subject to relevant resource constraints. These constraints can come in many forms, but are typically on 1) limiting thinking/computational time to reason, 2) limiting memory/space used to reason, 3) limiting the perceptual bandwidth or accuracy, and 4), limiting an agent's prior knowledge about the world, including its constituents and causal rules.

At a high level, the following alternate theories have been developed:

- *Bounded Rationality* [138]: Proposes a theory of rationality subject to constraints of relevance, like thinking time and memory.

- *Minimal Rationality* [27]: Seeks to identify the *minimal* set of conditions needed for an agent to be rational.

- *Ecological Rationality* [51, 50]: Extends bounded rationality to more closely consider the relationship between agent and environment; a decision is rational depending on the environment or context in which it is made.

- *Computational Rationality* [49, 92]: Formalizes bounded rationality through a mixture of computational models and artificial intelligence. Indeed, this is the most closely aligned to the RL variant I present.

In the following sections, I provide a birds eye view of each of these theories. Largely, I take their accounts to be correct. The nuances that differentiate them, while important, are not intended to be the focus of this work.

2.2.1 Bounded Rationality

Bounded rationality was first developed by Simon [138]. The core of the theory rests on the consideration of *constraints* placed on a given (would-be) rational agent: "Theories that incorporate constraints on the information processing capacities of the actor may be called theories of bounded rationality" (pp. 162 [138]). Simon's initial focus was in an economic context, so his discussion concentrates largely on practical rationality (though parallel conclusions can be drawn for epistemic rationality).

Naturally, as we will explore in the next section, there are several ways to constrain the information processing capacities of an actor. Simon suggests that bounded rationality more generally explores deviations from assumptions typically made by theories of rationality, including:

1. Variations of the utility function, for example, through the incorporation of risk/uncertainty.

2. Impose incomplete information: perhaps the agent must choose between options A and B while only being able to ask one question of limited scope about either A or B.

3. Deviations from the original goal.

Simon's attention was on practical rationality. As such, he develops a conceptual account of what it might look like to impose constraints on decision making. Simon raises the example of playing a game of chess, in which he defines two canonical problems: 1) choose an optimal sequence of moves that will win the game, or 2) come up with a means of accurately evaluating each move. These are, in essence, the two (overlapping) problems facing a chess playing agent. Morgenstern and Von Neumann [109] said of the matter, "...if the theory of chess (i.e. the complete tree of possible games) were really fully known there would be nothing left to play" (pp. 125). Simon recalls that they go on to suggest that, despite this striking fact, it does nothing to help guide an actual chess player:

> But our proof, which guarantees the validity of one (and only one) of these three alternatives [that the game must have the value of win lose or draw for White], gives no practically usable method to determine the true one. This relative, human difficulty necessitates the use of those incomplete, heuristic methods of playing, which constitute 'good' Chess; and without it, there would be no element of 'struggle' and 'surprise' in the game.
>
> –Simon (pp. 125)

This is precisely the problem of logical omniscience rearing its head once again – despite full knowledge of the rules of the game, it is a challenge to determine a decent solution to either of the two core problems Simon discusses. There is a striking similarity to playing chess and to Example 1. At best, we rely on heuristics based on many prior experiences of playing the game [37] (or solving logic puzzles). To actually solve the game requires searching a massive structure: Simon suggests there are roughly 10^{120} possible games, and so requires a great deal of thinking time to exhaustively search all possible games for the best move. This, and related matters, underlies our need for thinking about rational decision making in light of *some* constraints. How exactly to model the constraints, and how to conceptualize possible theories of rationality in light of those theories, has remained an open question.

Aumann [9] presented survey of several decades of work that have since built on Simon's ideas. Aumman concludes the piece by posing a question:

> We content ourselves with one open problem, which is perhaps the most challenging conceptual problem in the area today: to develop a meaningful formal definition of rationality in a situation in which calculation and analysis themselves are costly and/or limited. In the models we have discussed up to now, the problem has always been well defined, in the sense that an absolute maximum is chosen from among the set of feasible alternatives, no matter how complex a process that maximization may be. The alternatives themselves involve bounded rationality, but the process of choosing them does not.
>
> – Aumann [9], (pp. 12)

We return to this question later in the chapter by offering reinforcement learning as an appropriate model for inspecting rational behavior under realistic assumptions.

2.2.2 Minimal Rationality

Minimal rationality was proposed by Cherniak [27]; its main claim, like many of the competing theories we will discuss, is that ideal rationality is far *too* idealized to be useful to people:

> The unsatisfactoriness of the ideal general rationality condition arises from its denial of

> a fundamental feature of human existence, that humans are in the finitary predicament of having a fixed limit on their cognitive capacities and the time available to them.
>
> –Cherniak [27] (pp. 165)

On the basis of this background claim, Cherniak seeks out the minimal set of conditions needed of an agent to be considered rational. He takes this view to properly offer a normative account of rationality. The search for such an account leads to the minimal general rationality condition:

> If an agent has a particular belief-desire set, he would attempt some, but not necessarily all of those actions which are apparently appropriate.
>
> –Cherniak [27] (pp. 166)

This is in essence Cherniak's view: we need not on board all of rationality, only some smaller set of conditions that facilitate the right kind of belief forming or decision making practices.

2.2.3 Ecological Rationality

Ecological rationality was introduced by Gigerenzer [50], Gigerenzer and Todd [52] and Smith [146] and has received continued attention in the literature.[1] Like bounded rationality, ecological rationality concentrates on how agents should best be making decisions under more realistic assumptions than rational choice theory: "[We] propose a class of models that exhibit bounded rationality...These satisficing algorithms operate with simple psychological principles that satisfy the constraints of limited time, knowledge, and computational might, rather than those of classical rationality" (Gigerenzer and Goldstein [51], pp. 656).

However, unlike bounded rationality and its kin, ecological rationality focuses primarily on the relationship between agent and environment. The theory suggests that the essence of rationality is

[1] For more work in this vein, see Todd and Gigerenzer [165, 166], Gigerenzer and Goldstein [51].

tied up in a fundamental way to the environment an agent inhabits. Instead of rationality relying on the appropriate use of some (say, logical) tools, ecological rationality suggests that effective practical decision making is constituted by domain-specific heuristics that lead to quick and effective action.

For instance, Todd and Gigerenzer refer to the example of choosing to be an organ donor in Germany, where 12% of adults are organ donors, and Austria, where 99% are donors. Todd and Gigerenzer suggest that, when controlling for relevant other factors like economic status and cultural differences, we still lack explanation for why opt-in rates vary so much between. However, on further inspection, Johnson and Goldstein [72] find that the two countries differ as to the default setting: in Germany, individuals have to actively sign *up*, whereas in Austria, individuals have to actively *opt-out*. Johnson and Goldstein suggest the following heuristic is at play: "When faced with a choice between options where one of them is a default, follow the default." This example is intended to highlight how the choice of "institutional" aspects of the environment can clearly impact the behavior of individuals in a predictable way. Ecological rationality puts forth these environmental considerations as central: being rational is often about finding the appropriate heuristics to guide decision making, in light of the given environment.

One aspect of ecological rationality is pertinent to our broader discussion: in Chapter 4, I will investigate the role that concepts play in bounded rationality. Gigerenzer also articulates the importance of well chosen concepts (though he calls them "external representations") in the context of ecological rationality:

> Our argument centers on the intimate relationship between a cognitive algorithm and an information format. This point was made in a more general form by the physicist Richard Feynman. In his classic The Character of Physical Law, Feynman (1967) placed great emphasis on the importance of deriving different formulations for the same

> physical law, even if they are mathematically equivalent (e.g., Newton's law, the local field method, and the minimum principle). Different representations of a physical law, Feynman reminded us, can evoke varied mental pictures and thus assist in making new discoveries: 'Psychologically they are different because they are completely unequivalent when you are trying to guess new laws' (pp. 53). We agree with Feynman. The assertion that mathematically equivalent representations can make a difference to human understanding is the key to our analysis of intuitive Bayesian inference.
>
> – Gigerenzer [50] (pp. 94)

In essence, certain representations can be more conducive to the right kinds of thinking, even if they are mathematically equivalent. I will ultimately agree entirely with the point articulated here by Gigerenzer and Feynman, but the particulars will differ slightly.

2.2.4 Computational Rationality

Computational rationality, like the others described, posits an alternate form of rationality, this time with the theory of computation at its center [92, 49]. Two slightly different variants of the theory have been proposed, the first by Lewis, Andrew, and Singh [92], the second by Gershman, Horvitz, and Tenenbaum [49].

On the Lewis, Andrew, and Singh view, the focus is on *Optimal Program Problems* (OPP), which take three things as input: 1) and environment, 2) a resource-constrained machine, and 3) a utility function. The study of computational rationality, on their view, is the study of which methods solve these OPPs; by framing things in this view, the theory is uniquely positioned to ask and answer questions of the form: "What should an agent with some specific information-processing mechanisms do in some particular environment?" (pp. 305). In many ways, this view is well aligned with what I will go on to propose in the next chapter: I will simply pose this question through the lens of RL.

Separately, Gershman, Horvitz, and Tenenbaum propose computational rationality as a mixture of AI formalisms with typical studies of rationality. Like the other theories so far discussed, one of the aims is to escape ideal rationality, this time by grounding resource constraints in terms of both computational time and space *and* knowledge of the environment through some the tools of AI.

One of the main considerations made is the careful monitoring of the cost of expending one's resources. That is, suppose we give agents awareness of the *cost* inherent in any computation they were to execute. The resulting notion of rationality would be one that only uses computation when necessary, optimizing based on these costs. Such processes have been studied at length in other recent AI and cognitive science literature, as by Zilberstein [176, 177, 178], Russell and Wefald [130], and more recently by Griffiths et al. [56], Lieder et al. [96].

Gershman, Horvitz, and Tenenbaum offer the following summary of the framework:

> Computational rationality offers a potential unifying framework for the study of intelligence in minds, brains, and machines, based on three core ideas: that intelligent agents fundamentally seek to form beliefs and plan actions in support of maximizing expected utility; that ideal MEU calculations may be intractable for real-world problems, but can be effectively approximated by rational algorithms that maximize a more general expected utility incorporating the costs of computation; and that these algorithms can be rationally adapted to the organisms specific needs, either offline through engineering or evolutionary design, or online through meta-reasoning mechanisms for selecting the best approximation strategy in a given situation.
>
> – Gershman et al. [49], (pp. 278)

I here add my own support of the generality of their proposal: there is much to gain by using a general framework. These three core ideas closely approximate many of the properties I will attend to in arguing for RL as a setting for discussing bounded rationality, though I spend significantly less time on this second property (that of metareasoning about incorporating the cost of computation). Instead, I will go on to argue that agents should form the right concepts for making planning easy. Though, again, the difference in these points is relatively subtle.

Going forward, I will not concentrate on explicit distinctions between the above theories. I take them all to be highlighting important issues with ideal rationality, and each bring something important to the table. Instead, I next focus on unpacking some of the tools at our disposal that can use give precise meaning to placing constraints on rational agents, including how to limit an agent's thinking time, memory, and amount of evidence available.

2.3 Computational Complexity

One of the two central claims of bounded rationality is that imposing constraints on agents is a helpful practice. To explore this point further, we need some notion of "constraint". We here explore models for formalizing the relevant constraints: thinking time, thinking space, and amount of evidence an agent has collected thus far.

The first constraint imposes limitations on the amount of thought or introspection that can go into reasoning about a particular matter. Given the appropriate amount of time to think (and perhaps the right tools, like a chalkboard), agents may come to have the correct beliefs or choose the right decisions. Even then it's reasonable to suppose that, due to resource constraints, many agents may make a mistake somewhere along the way.

To codify this notion we turn to the computational difficulty of solving problems of a particular kind. We will find that the straightforward translation of the existing theory fails to fully capture what we aim for when we constrain rationality from a computational perspective. Many alternatives and extensions to this theory do exist, however, and we will explore whether these extensions' can play the right kind of role in rationality, too.

At a high level, the desirable aspects of computational theory are as follows:

1. A simple, general method for characterizing the difficulty of certain kinds of problems.

2. We can prove bounds of a variety of forms, for a given problem:

   (a) *Lower:* What are the *minimum* number of primitive moves needed to solve a given problem?

   (b) *Average:* On average (across problem instances), what is the number of primitive moves needed to solve a given problem?

   (c) *Upper:* After how many primitive moves can we guarantee that we will have solved any instance of the problem?

However, there are glaring weaknesses to the theory, too:

1. There is no obvious cutoff for what constitutes a "reasonable" bound. Some have argued in favor of different thresholds, such as the polynomial worst-case boundary, but it is unclear this (or other choice of boundary, in principle) is the right answer.

2. The most significant result of the area, $P \overset{?}{=} NP$ is still unproven, and many critical results are a direct consequence of whether this result is True or False.

3. Some may take issue with translating results from an ideal *computer*, to an ideal *reasoner*, since it is natural to suppose that people cannot reason purely in terms of abstract symbols.

4. There is no natural notion of a "primitive move".

We discuss these and others in more detail shortly. The takeaway is that we can defer to existing analysis on the difficulty of solving certain types of problems as a means of refining our notion of a realistic agent.

2.3.1 The Theory of Computation

The central objects of study in the theory of computation are *computational problems*, and the *algorithms* that solve them. The space of problems can be broken down into a few subcategories, with the most canonical case being a decision problem:

Definition 1 (Decision Problem): *A* **decision problem** *is a yes-or-no question asked of an input set: $x \in \mathcal{X}$. For example, a problem may be defined as being asked of all inputs $x \in \mathcal{X}$, the question: "is x a prime number?".*

Further problem types generalize decision problems by searching for satisficing instances of a particular predicate (search problem), or finding an element that maximizes or minimizes some function (optimization).

For instance, given a natural number $x \in \mathbb{N}$, the problem of determining whether x is prime or not is a decision problem. The solution, called an algorithm, endeavors to solve this problem *for all inputs* of the relevant kind (all natural numbers, in the case of primality).

Why bother defining problems in such a specific way? Well, with the specifics nailed down, we can then analyze the limitations of problem solving. We gain access to a type of mathematical result we didn't have access to previously: the characterization of a problem's difficulty, from the perspective of how many mental moves (or how much memory) it takes to solve. Is primality testing *harder* than sorting a deck of cards? What is the fastest possible strategy for routing traffic in a city, optimally? How hard is it to find the shortest proof of a particular theorem? If we can answer these questions, we start to gain access to problem solving tools (for real problems) we didn't have before.

It is not too big of a jump to see how we might then use these constraints to start talking about rational behavior under realistic assumptions. We can fix the problem of logical omniscience by only considering agents that have access to N computations, which means they can only solve problems whose fastest solution takes no more than N steps. This helps us both 1) guide decision making (what should I do given that I can only run N computations?) and 2) place blame (did that person misstep, given that they only had N computations? Or was it just due to their resource constraints?). How might these constraints be instantiated? We have a few options, with the most typical being worst case complexity.

2.3.2 Worst Case Complexity

There are a few reasonable interpretations of "how many mental moves it takes to solve". First, and the most traditional to complexity theory, we define problem difficulty in terms of the number of primitive computations needed to solve the *hardest* instance of an input of size N. Again, the field jointly focuses on the memory resources needed to solve certain problems – we will often talk of either *time* resources or *space* resources required to solve certain problems. This is the essence of worst case complexity.

Example 2. *Suppose we are given a list of words, L. We know the list contains N words. The problem we'd like to solve is whether a given word, w, is in the list.*

How might we write down a sequence of steps that will work for any given L and w?

From the perspective of a person, if N is sufficiently small, we might be able to see the entire list all at once. So we just look.

Suppose N is arbitrarily large, though. Now what might we do? A natural strategy would be to start at the first element of the list, ℓ_1, and check whether $\ell_1 \overset{?}{=} w$. If it does, we know $w \in L$. If not, we continue to ℓ_2, and so on down the list. If we check ℓ_N and still haven't found w, then we know w is not in the list. Note that this is guaranteed to work for any list, for any word.

Question: How many "mental moves" does it take to run the above strategy? It all depends on how we define a "mental move"! In Chapter 4, and indeed, one of the focuses of this thesis, I argue that concept choice is intimately connected to an agent's rationality; this is our first hint as to why concept formation is so important to decision making. If we consider checking an entity $\ell_i \overset{?}{=} w$ as a move, then *in the worst case*, we need N moves. Why? Well, if the item is not in the list, we have to check all N positions. Alternatively, if we consider comparing each letter between our word and a given word in the list, then we have to execute at most $N \cdot M$ operations, with M the length of our word.

In this example, we see that there exist lists for which we *must* take N moves in order to solve the problem in a satisfactory way. This is known as *worst case analysis.* For a given input of size N, for a given computational problem $\mathscr{P}$, what are the most number of mental moves we'll need in order to solve $\mathscr{P}$?

One caveat to the theory is that the field of computational complexity usually doesn't care about the precise details of the number of mental moves: a problem that takes N vs. a problem that takes $N + 1$ moves are said to be of effectively identical difficulty. More specifically, for a function $f(N)$, with N the size of the problem (more on that shortly), complexity tends to group problems of certain difficulties based on the dominating term of $f(N)$. So, if one problem is known to take at worst $f(N) = 2N + 1$ operations, complexity theory only cares about the fact that the N term dominates as N goes to infinity. Consequently, most complexity results will tell us the rough order of magnitude of growth as the input size grows arbitrarily large, but will ignore details. So, for a few examples:

$$f_1(N) = 3N^2 + N + 7 \approx N^2 \tag{2.10}$$

$$f_2(N) = e^N + 7N^2 \approx e^N \tag{2.11}$$

$$f_3(N) = 8N\log(N) + 7 \approx N\log N \tag{2.12}$$

In general, the worst case complexity is defined as follows:

Definition 2 (Worst Case Complexity): *The* **worst case complexity** *of solving a given computational problem $\mathscr{P}$, with input of size N, is said to be the dominating term of the number of primitive computations needed to solve the hardest instance of size N.*

"Dominating term" here describes the asymptotic behavior of the number of computations needed. If our algorithm takes $3N^2 + N + 1$ computations to solved a given problem, as N goes arbitrarily large, the terms N and 1 will have a negligible effect on the overall number of computations (even the multiplicative factor of 3 will, too!). Consequently, we would call $3N^2 + N + 1$ an $O(N)$ algorithm. The notation $O(\cdot)$ is called "big-oh-notation" and simply expresses shorthand for defining the asymptotics of the number of computations required (see Sipser [144] or Arora and Barak [7] for more background).

Using this definition, the field of complexity theory has mapped out the territory of the difficulty of many known problems according to their worst case complexity. The famous classes of P and NP are those classes of problems that have worst case scenario that is at most a polynomial of

N (P), and those problems for which candidate solutions can be verified in at most a polynomial number of operations (NP) [32].

Shortcomings of Worst Case Complexity

Let's take a step back. We've seen how we can explicitly spell out the hardness of a problem in terms of the number of mental moves needed to solve the problem's hardest instances. If we return to our original question of practical rationality, we now have a clear mechanism for avoiding logical omniscience: if an agent is given some computational budget, we can be clear about which problems that agent can in principle solve, and which they cannot.

But, there are a few philosophical issues latent in the above theory that require attention.

What Counts as a Computation? First: what constitutes a "mental move"? In the context of complexity theory, there is at least one useful variant of "mental move": the number of primitive operations needed by a Turing Machine. What, then, in the context of reasoning should count? A natural response might be to look to the psychological literature to characterize our "atomic" mental moves, and give an account of our cognitive constraints based on these atoms. But, these mental moves may not be universal across different agents, cultures, languages, or areas of study. In general, it seems we lack a clear sense of a mental move.

Why Worst Case? Another natural objection to treating worst case analysis as the appropriate form of assigning problem difficulty is that the problem might be extremely easy, except for one pathologically difficult instance. For the many iterations of rationality we might care about, worst case is *far* too strong. Most problem instances we encounter in our world tend to adhere to a particular type of structure – it seems silly to measure our ability to reason based on how long it would take us to solve problems that *never* emerge in our world. We want our constraints to be realistic, not needlessly attached to some far off possible world.

Problem Size A necessary component in assessing a problem's complexity is measuring the size of the input. For physical agents, there is no such obvious measurement. Physical agents do not take as input crisp symbolic objects. We do not get to observe lists in the form: "$\ell_1, \ell_2, \ldots, \ell_n$". All inputs are processed through noisy, uncertain sensors, that only given partial information about our surroundings. How do we characterize the size of a problem? One might consider measuring the bandwidth of our sensory mechanisms, but even still, as per the phenomena of Change Blindness [124], we surely do not process every piece of information our sense organs can perceive. Instead, we throw away information and distill what we perceive into something more malleable.

The Polynomial Boundary As alluded to previously, it is not obvious what to define as the right computational constraint, even if we are willing to take on board worst-case analysis. A common take in the complexity community is to assume that agents can be given a polynomial amount of resources to solve its problems – so, for a problem on input of size N, the agent in question must be able to solve the problem of relevance in N^c, for some constant $c \in \mathbb{R}_{\geq 0}$. For example, modern cryptography (that underlies many of our security systems) typically assumes that cryptographic defenses must be robust with respect to a polynomial adversary – if a malicious onlooker can crack your code using only a polynomial computational budget, then your code is said to be insecure. However, it is not obvious why polynomial should be the appropriate boundary when discussing rationality. Imagine you are placed in a room with a whiteboard, access to the internet, and plenty of coffee, tea, water, and snacks. You have 10 hours to solve a given problem. How you align your beliefs after this highly focused experience compared to someone driving quickly by a park that must update their beliefs about the number of trees in the park should depend entirely on the situation at hand: how much time, perceptual fidelity, and data do you have available to you? Moreover, there clearly exist polynomials with large enough exponents that prevent us from performing the relevant number of computations in a single lifetime (or, exponents with a small base). It is a striking fact that most known polynomial time algorithms have relatively small constant factor exponents, and likely not one we should rely on to justify the polynomial boundary as the definitive line-in-the-sand for reasonable computation.

Exactness vs. Approximation The traditional definition of worst case complexity tolerates *no* notion of error. The problem must be solved, exactly, for all possible inputs. This is an extremely high bar, given that some error is practically necessary for effectively any agent – a theory of rationality that expects perfection is unhelpful when agents will necessarily make mistakes. More generally, we might only care about getting an answer that is sufficiently close to the true answer. For example, when asked how much gas we have in the car, we only really need a rough estimate. Approximation algorithms give us a formal means of capturing this notion: algorithms are allowed to return an answer that is said to be sufficiently close to the true answwer (often with high probability).

In summary, we find five shortcomings of treating worst case complexity as the appropriate means of constraining a physical agent's resources in deliberation:

1. *What Counts:* It's not obvious what should constitute a primitive move in deliberation.

2. *Problem Size:* When our inputs are coming from sense organs, there is no clear measurement for problem size. So, we may lose the ability to talk about deliberation constraints.

3. *Why Worst Case:* Only considering the worst case ignores the fact that we might never have to actually deal with the worst case.

4. *Constraints – Polynomial, or Otherwise:* The polynomial boundary is not clearly the right choice for constraining physical agents.

5. *Exactness vs. Approximation:* Surely we can tolerate occasional error, or near-optimal results. So, hardness-of-approximation seems a more appropriate tool for analysis.

I next turn to two alternatives for that may help us alleviate some of the above five concerns.

2.3.3 Average Case Analysis

Average case analysis extends worst case analysis by supposing we have some collection of possible inputs, which we tend to describe via a probability distribution over our input space, D [91, 18]. That is, for each element of interest, $x \in \mathcal{X}$, D assigns some probability to each element $D(x) = \Pr(X = x)$, indicating the likelihood of the element x occurring. Then, our interest is in assessing the *expected* number of computations needed in expectation over the distribution D. This gives us a way to avoid anomalous and pathological cases that drive up the complexity of our certain problems – we simply suppose that in our world, such instances might have negligible or zero probability.

For instance, if we return to our example of searching for an element in a given list, we might supposed that we place *all* of the probability mass on problem instances on cases where the item is either 1) in the first half of the list, or 2) not in the list at all. Then, we can see the complexity will be roughly $\frac{N}{2}$, for a length N list.

As we learn more about our world, we can better characterize the distribution D for different domains, thereby tightening our knowledge of how hard certain problems should be for agents in our world.

Average case complexity gives us an additional tool to constrain problem difficulty, but many of the issues of worst case analysis persist. The most important one, perhaps, being that whereas before we didn't know how to measure problem size, now we don't know how to pick D *and* measure problem size. For agents wandering around the Earth it is unclear how to model the distribution of problems they will encounter. So, we can't really gain insight into how hard, on average, the problems they face will be, unless we know before hand where the agent will go. Coming up with such a distribution is arguably as hard as coming up with a perfect predictive model of the world ("can we predict what problems we'll see?"). If we're wrong about D, then we can be wrong about measuring complexity, too.

So, while Average Case gives us further granularity at specifying constraints, it still doesn't get everything quite right.

2.3.4 Approximation

Moreover, it seems necessary to allow people to error. Physical agents will necessarily be uncertain about aspects of their worlds about which they will reason, or will need to make split second decisions about complicated matters. In either case, sometimes, *error is permissible from a rational perspective*. For instance, again consider the list case: we have a list of size N, and we search for an element e. But, we only get $\log N$ seconds to make the decision. How best should we spend this time? What would the ideal rational agent do *subject to this time constraint*?

In general, we need some scheme for accommodating error. When we focus on the learning theoretic framework in the next section, we will see how statistical considerations will produce a natural notion of error tolerance. But, from a computational perspective, it turns out we can get a similar, well justified notion of error.

This view of error comes from the field of approximation algorithms [71, 171]. The goal here is to develop an algorithm, A, that nearly solves a problem $\mathscr{P}$, such that the solution generated by A is *sufficiently close* to the true solution. Consider the following example:

Example 3. *Consider the knapsack problem, $\mathscr{P}_{ks}$, in which we are given a sack with weight capacity C, and a collection of N objections $o_1, \ldots, o_N$. Each object has both a weight and a value.*

The goal of the problem is to fill the sack with some subset of objects that maximizes the total value of objects in the sack while not exceeding the weight capacity. So:

$$\max_{O \subset o_1, \ldots, o_N} \sum_{o \in O} V(o), \tag{2.13}$$

$$s.t. \sum_{o \in O} W(o) \leq C. \tag{2.14}$$

With $V(o)$ expressing the value of object o and $W(o)$ expressing the weight of object o.

This problem is a classical NP-Hard problem (so we don't think it can be solved in polynomial time).

However, it can be approximated quite well with the following strategy. Per the result of Dantzig [36], we can simply sort the items based on their value-per-unit-weight, and add the most value-dense items until we reach capacity.

Such a strategy is guaranteed to achieve a bounded degree of error (under certain assumptions, which we here ignore). If the true optimal solution achieves a total knapsack value of m, then the above strategy will fill the bag with no less than $\frac{m}{2}$ value. Pretty good considering how simple (and easy to run) the strategy is!

So, let us take stock. Some problems are hard to compute, whether from a worst-, average-, or even best-case perspective. However, we can still come up with strategies for solving these problems in a way that guarantees reasonable outcomes, given less resources. What are the philosophical implications? The main upshot is that even the ideally rational agent, *subject to computational constraints*, can achieve reasonable solutions in the time allotted.

Still, the shortcomings discussed previously do not permit a fully general means of applying computational constraints to a reasoner. In the approximate case, a further difficulty appears: what is considered tolerable error?

And all the while, our problem size is not measurable. This is perhaps the most damming of the remaining difficulties. For an arbitrary problem encountered by a physical agent, how are we to measure its size? In principle, many problems can be compressed and solved, too. Do we measure the perceptual bandwidth of the agent? The memory? The size of the mental atoms needed to think about the problem? The size of the smallest compressed form of the problem? We lack the appropriate tools to capture the appropriate measure. We will return to this matter in Chapter 4, where we explore the relationship between *concept formation* and *efficient use of cognitive resources*: with the right concepts, certain computations become easier. So, the ideally rational physical agent forms concepts that make computational the most robust and efficient possible, while still retaining accuracy.

2.4 Computational Learning Theory

We have now seen how to characterize the difficulty of a given computational problem. Still, there is much more to rationality than deduction. What about induction? How do we deal with uncertainty? What about perception, and the appropriate gathering or handling of evidence? Questions of this nature find a better home in the computational *learning* framework, rather than standard computational theory. There are of course computational problems of relevance to discuss: given some data, and a set of initial beliefs, we might ask about the worst/average/best case complexity of updating one's beliefs to best reflect the current state of affairs, in light of the evidence. While such questions should make an appearance in our discussion of rationality, we still require an additional formal tool to make progress in the more general picture.

This tool is that of learning theory, often called either Computational [168] or Statistical [170] Learning Theory. Together, they will offer further techniques for cleanly defining rationality.

At a high level, they give us the ability to characterize how much evidence is needed before an agent can acquire a reasonable hypothesis of some property or state of affairs. In this way, there is a parallel between these theories and Bayesian views of belief updating. The theories depart from Bayes in three critical ways: 1) We'd like to characterize the *precise* amount of evidence needed, in the same way we previously showed we can define the number of computations needed, 2) We typically require that all belief updating must be done with a reasonable computational budget, and 3) Typically, the results are agnostic to priors, though there are natural extensions to incorporate Bayesian methods of belief updating, proposed by McAllester [104].

We will discuss two frameworks, the Probably Approximately Correct (PAC) Learning framework introduced by Valiant [168], and the Statistical Learning Theory framework, with the earliest results presented by Vapnik and Chervonenkis [169]. Each give us a mathematical model for discussing the relationship between quantities of evidence and accuracy of beliefs.

2.4.1 Probably Approximately Correct Learning

The PAC framework focuses on the following question: *how much evidence does an agent need to have appropriately learned a given boolean function?*

Here, “evidence” means a pair of datum $(x, c(x))$, appropriately learned means “has a good enough understanding of the function” ($h \approx c$), and “boolean function” means some labeling of entities in a given domain ($c : \mathcal{X} \rightarrow \{0, 1\}$). Typically, these boolean functions are called “concepts”, but they are qualitatively quite different from how we will use concept in later chapters.

More formally, we suppose we’re targeting a particular kind of item – let’s say, for example, that we’re dealing with images. A boolean function is a labeling for each image in our collection. For example, one concept might be “landscape” that picks out which images are landscape photos and which are not. In the simplest case, we define our collection of objects according to a (possibly infinite) alphabet, $\mathcal{X}$. The function, then, is $c : \mathcal{X} \rightarrow \{0, 1\}$ that separates entities in the collection into the objects that satisfy the property and those that don’t. So, if $c(x) = 1$, then x is a landscape photo, and if $c(x) = 0$, then x is not a landscape photo.

Naturally, this understanding of “function” is much too simple. Actual functions of interest can be far more nebulous than we have just given them credit for. So, we will acknowledge that in the PAC setting, our notion of function is perhaps too sparse, and attend to this point more fully in Chapter 4. Other parts of learning theory have generalized this notion to multi-class ($c : \mathcal{X} \rightarrow \{0, \dots, n\}$), and real-valued function ($c : \mathcal{X} \rightarrow \mathbb{R}$), but these complicate matters.

As a final assumption, we suppose the world is endowed with an infinitely wise and patient teacher that will label any object we select from the collection with the true label. More formally, we suppose that the agent in question can select entities from the collection according to some probability distribution D (that is, the support of D is $\mathcal{X}$). When the agent samples an entity

$x \sim D$, the teacher tells the agent the true label of the concept. So, at each time step, the agent samples x and observes the pair $\langle x, c(x) \rangle$. For example, in the case of the images, the agent gets to see an image, and is told the image either does or does not contain a dog.

The central problem is as follows. Given a collection of n independently sampled and identically distributed (i.i.d.) pairs, $\langle x_i, c(x_i) \rangle$, for what value of n can the agent select a *hypothesis* h from a given hypothesis space $\mathcal{H}$ that roughly matches the original function? The hypothesis space in the most general case is simply equivalent to the space of functions with the same mapping as the function, so $\mathcal{H} = \{h : \mathcal{X} \to \{0, 1\}$, but this need not be the case. We might suppose a *simplicity* bias on the hypothesis space, that restricts $\mathcal{H}$ to functions that are suitably simple (perhaps as measured by their computational complexity!).

Our mission is to measure the effect of the amount of evidence on the quality of an agent's understanding of different functions. In this way, it resembles Bayesian belief updates. The major difference is to prove theorems that relate the size of evidence with the confidence an agent is justified in having with respect to a particular function. These theorems will vary slightly depending on whether the learning problem is realizable or agnostic, but the takeaway is the same: an agent needs N data points to adequately learn a hypothesis that closely resembles c.

Consider the following example concentrating on learning an interval in 1D space:

Example 4. *Let c denote the boolean function $c(x) \equiv 5 \leq x \leq 20$, with x taking values in natural numbers from 1 up to 100. Let D be a uniform distribution over the interval $\mathbb{N}_{[1:100]}$.*

If we choose our hypothesis space to be the set of **all** functions that map all natural numbers from 1 to 100 to $\{0, 1\}$, we have a massive space of functions to search through: for each x, we can either assign it to a 0 or 1, and thus, there are a total of 2^{100} possible hypotheses. That is monstrous!

In an ideal learning setting, we can take advantage of some domain knowledge, or use a generally useful inductive bias over model selection such as a simplicity prior. For instance, suppose we search for a single decision boundary, τ, where everything greater than or equal to τ is in the interval (assigned to 1), and everything below τ is outside the interval is set to 0. This yields only 100 hypotheses (one for each possible setting of τ), down from 2^100. But note that now we are in the agnostic setting – even the best boundary of $\tau = 100$, we will error 0.15 of the time. So, this new hypothesis class is easier to search through, but less expressive. This is a fundamental dilemma underlying learning, and one that will be critical to our later discussion.

How can we balance between these two extremes? In general, if we don't know anything about our domain, we cannot. Fortunately it is often the case that certain kinds of inductive biases will be natural for a variety of domains.

What we're after is a hypothesis class that is sufficiently general, but not so general as to blow up the size of the hypothesis class.[2]

For the interval example, let us suppose we consider the space of hypotheses that define any continuous interval in $\mathbb{N}_{[1:100]}$:

$$\mathcal{H}_{\alpha,\beta} = \{h(x) = \mathbb{1}\left(\alpha \leq h(x) \leq \beta\right)\}, \tag{2.15}$$

for $\alpha, \beta \in \mathbb{N}_{[1:100]}$, and $\alpha \leq \beta$.

Now, after even just one sample, we can already start to eliminate huge portions of our hypothesis space. So, this is a suitable choice for $\mathcal{H}$ – it is not too big, allows quick learning, and can faithfully represent the true function. Learning is entirely about making precisely these trade-offs.

The primary results in the PAC literature are *generalization error bounds*, which effectively

[2] This trade-off is intimately connected to the bias-variance trade-off [48].

suggest how much evidence is needed by the agent until it can pick a hypothesis, $h \in \mathcal{H}$, such that the chosen hypothesis will not differ too much with respect to the true function. Here, "differ" is measured in terms of a bound on the probability of error, with respect to samples taken from the distribution. Such results will (sort of) extend to the full RL setting, allowing us to relate the amount of evidence an agent needs before it can adequately make justified decisions.

Definition 3 (PAC Learnable): *A boolean function c is said to be* **PAC Learnable** *if, for a given hypothesis class $\mathcal{H}$ such that $c \in \mathcal{H}$ (realizability), then there exists an algorithm that will output an $h \in \mathcal{H}$ that is sufficiently close to c.*

"Sufficiently close" has a precise technical meaning, but it isn't all that important for our present discussion. For completeness, a boolean function is PAC Learnable if, for a given $\varepsilon, \delta \in (0, 1)$, the *loss* of the chosen hypothesis $L(c, h)$ is bounded with high probability:

$$\Pr_{x \sim D} \{L(c(x), h(x)) \leq \varepsilon)\} \geq 1 - \delta. \tag{2.16}$$

The δ parameter captures the "probably" part of the **P**AC acronym, while the ϵ captures the "approximately" part. The loss function will vary depending on the family of boolean function and hypothesis. If we are learning a boolean function, then a natural choice for loss is the mean squared error:

$$L(c(x), h(x)) := |c(x) - h(x)|^2. \tag{2.17}$$

If we are instead learning a probability distribution (so our hypothesis space is the space of probability density functions), then a more natural measure would be any probability distance metric.

Basically, the PAC property is said to obtain of an algorithm if, when the agent outputs a hypothesis, the hypothesis is guaranteed to be pretty close to the true function with high probability.

This is a strong guarantee for an algorithm to have!

Can we make this guarantee of people? That is, can we state that others in our community use concepts in a way that is largely consistent with each other (where "consistent" means, with high probability, they are ε apart?) Probably not. In large part because our data is not independent and identically distributed (we have routines, and tend to have certain experiences in sequence), we all perceive things differently according to our representational or perceptual biases (is the dress blue or gold?), and our concepts are probably not so rigid so as to afford such precise guarantees (is the pile of sand a heap or not?).

Still, the PAC learning frameworks gives us extremely general results about boolean function learning:

Theorem 1. *Every boolean function class is PAC learnable under a finite Hypothesis class, $\mathcal{H}$, (and the realizability assumption that $c \in \mathcal{H}$) with evidence of at most:*

$$\left\lceil \frac{\log \frac{|\mathcal{H}|}{\delta}}{\varepsilon} \right\rceil, \tag{2.18}$$

where $\delta, \varepsilon \in (0,1)$ denote accuracy parameters.

This states that if we have finitely many hypotheses to consider before finding a sufficiently good one, we need roughly $\log(|\mathcal{H}|)$ labeled data points before we can learn the function of relevance.[3]

When we return to our discussion of concept selection in Chapter 4, it will be important to clarify how concepts impact quantities like the above generalization error bounds – can an agent's choice of using a group of concepts C_1 guarantee that the agent will be more able to correctly make

[3] One might wonder what happens when the hypothesis class is not finite – Vapnik and Chervonenkis [169] introduce the Vapnik-Chervonenkis Dimension (VC-Dimension) that allows careful measurement of a hypothesis class' complexity, even if its size is infinite. Then, the evidence needed to learn most functions depends on the VC-Dimension of the function class, *not* its size.

inductive inferences than those represented by a different group of concepts C_2? Simply: yes! To gain some initial insight into why this may be the case, let us turn to one of the seminal results of learning theory. The result lets us clarify the nature of errors made by different approaches to learning.

Specifically, there are said to be two sources of error 1) approximation error, and 2) estimation error. Approximation error occurs relative to concept c if the agent is considering a set of hypotheses, $\mathcal{H}$, such that the *best* hypothesis in $\mathcal{H}$, which we denote h^*, is still sufficiently distant from c. So:

$$\varepsilon_{approx} := \min_{h \in \mathcal{H}} L(h, c), \tag{2.19}$$

for some function that measures the *loss* of a chosen hypothesis, h, relative to c, and perhaps relative to a body of evidence, E. Thus, if c is not contained within $\mathcal{H}$, then the approximation error is the gap between the best possible hypothesis the agent *could* learn, and the concept.

Why would we ever restrict the hypothesis class to not contain c? It turns out there are a few reasons. The primary one being that a smaller space of hypotheses is easier to search through. In many ways, restricting the hypothesis space amounts to injecting priors about the world, also called inductive biases [107]. Something like a "simplicity prior" that places preference on simpler hypotheses rather than complex ones might, in a sharp form, trim the hypothesis space to consist solely of the simple hypotheses.

The second source of error is called estimation error. This effectively amounts to error that results from noise in the learning process itself:

$$\varepsilon_{estim} := \mathbb{E}_{E \sim D}\left[L(\hat{h}_E, h^*)\right], \tag{2.20}$$

where the expectation is taken over the sampled evidence from the data distribution, D, and $\hat{h}_E$ is the chosen hypothesis by a particular agent based on evidence E.

Example 5. *Suppose an agent is trying to learn the bias of a coin, q. Before collecting any data, the agent restricts the hypothesis class to consist of the hypotheses that the coin has bias either 1.0, 0.8, 0.6, 0.4, 0.2, or 0.0. So:*

$$\mathcal{H} = \{1.0, 0.8, 0.6, 0.4, 0.2, 0.0\}. \tag{2.21}$$

We further suppose that the loss function in this case is simply the absolute difference between q and c, $L(q, c) = |q - c|$.

The agent then flips the coin n times by sampling from Bernoulli($\theta = q$).

Before even observing the data, we know that if the true function (in this case the true bias of the coin), is *not* contained in $\mathcal{H}$, we can get an upper bound on the approximation error:

$$\varepsilon_{approx} \leq \max_{c \in [0:1]} \min_{h \in \mathcal{H}} L(h, c) \leq 0.1. \tag{2.22}$$

Naturally, we can tighten this bound as we know more about the true function. For example if $q = 0.45$, we know that ε_{approx} is 0.45.

To determine the estimation error, we need to inspect how the agent chooses its hypothesis, $\hat{H}$, and we'd need to better understand how the evidence will be distributed. In essence, the agent will be presented with some series of n coin flips, $\{T, H, H, T, H, T, T, \dots\}^n$, and will be asked to choose a hypothesis $\hat{h}$. Two natural estimators are the maximum likelihood estimator (MLE) and maximum a priori estimator (MAP); in the MLE, Bayes rule is used to determine the hypothesis that maximizes the likelihood of the evidence, in the MAP, a prior is chosen and the hypothesis that maximizes the posterior probability of the given evidence. So, letting $E = \{T, H, \dots, \}^n$ denote the body of evidence:

$$MLE(E) = \max_{\hat{h} \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \log \Pr(e_i \mid \hat{h}). \tag{2.23}$$

$$MAP(E) = \max_{\hat{h} \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \log \Pr(e_i \mid \hat{h}) \Pr(\hat{h}). \tag{2.24}$$

Depending on choice of prior, $\Pr(\hat{h})$, the MAP estimator will make different decisions from the MLE. Choice of estimator, then, places preference on different kinds of properties of the solution; do we care about the impact of a prior? Do we care how accurate the estimator is after n data points for small or large n? How about in the limit of data?

These are precisely the considerations that determine the trade-offs that go into determining ε_{estim} relative to a particular choice of learning method (in the above, a simple statistical estimator).

From the above example, we can see how, subject to different constraints or different knowledge of a particular problem, agents will be presented with a variety of choices about how to make decisions. Through the framework of learning theory, we can start to shed light on how these kinds of behaviors affect one's capacity to retain accurate beliefs.

As with computational complexity theory, there are several shortcomings to the model to treating it as a perfect way to measure a statistical constraint:

1. Experiences are not independent and identically distributed. As we move around the world, the evidence we're likely to see changes. Most of learning theory makes the assumption that there is a fixed distribution over experiences, D, the agent draws data from.

2. Experiences are not nicely "formatted" into easily interpretable properties. As with complexity theory, agents do not get to observe the flip of a coin as either T or H, but instead receive high dimensional and noisy sensor input of the results of a coin flip.

3. A proper understanding of concepts likely requires more than just boolean functions.

Each of these points is potentially crippling to the usefulness of the theory. But, revisions exist that avoid many of the core concerns. In fact, we will find that RL has at least a partial solution to these problems: experiences will not be i.i.d., experiences will be an uninterpretable mess to begin with, and our concepts will be arbitrarily rich functions.

To summarize, computational learning theory gives us a rich formal framework for analyzing how hard it is to learn certain boolean functions. A featured assumption of RL removes prior knowledge from the ideal rational agent – how *should* an agent learn, rationally? Which trade-offs should an agent make? To understand how to answer these questions, we will need formal tools like the methods introduced above. We can now cleanly state the relationship between an agent's available resources (in this case, evidence of the form of $\langle x_i, c(x_i)\rangle$ pairs), and its capacity to learn.

2.5 Reinforcement Learning

Intuitively, RL defines the problem of an agent learning to make good decisions in an environment through interaction alone. The primary objects of study of RL are computational agents, the worlds they inhabit, and interactions thereof. An agent is any entity capable of taking action, perceiving relevant information about its surroundings, and receiving rewards that indicates the present utility inherent in the current state of the world. More precisely, the RL problem is defined as follows:

Definition 4 (Reinforcement Learning Problem): *An RL agent interacts with a world via the repetition of the following two steps:*

1. *The agent receives an observation o and a reward r.*
2. *The agent learns from this interaction and outputs an action, a.*

The goal of the agent during this interaction is to make decisions so as to maximize its long term received reward.

What, then, does the "world" look like? In the psychological literature, the world can be any manner of phenomena that people (or other animals) experience; how we learn language, games, norms, or otherwise. Traditionally, in computational RL, the world is assumed to be modeled as a Markov Decision Process (MDP) [122], a convenient formalism for describing sequential decision making problems. An MDP is defined as follows:

Definition 5 (Markov Decision Process): *A* **Markov Decision Process** *is a five tuple:*

- $\mathcal{S}$*: a set of states describing the possible configurations of the world.*
- $\mathcal{A}$*: a set of actions, describing the possible choices available to an agent.*
- $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$*: a reward function, describing the task to the agent.*
- $T : \mathcal{S} \times \mathcal{A} \rightarrow \Pr(\mathcal{S}))$*: the transition function, which denotes the probability of arriving in the next state of the world after an action is executed in the current state.*
- $\gamma \in [0, 1)$*: a discount factor, indicating an agent's relative preference between near-term and long-term rewards. As γ gets closer to 0, agents prefer near term reward more.*

The "Markov" in **M**DP indicates that the transition function, T, and reward function, R, both only depend on the current state of the world (and action), and *not* the full state history. So:

$$\Pr(s_{t+1} \mid s_t, a_t) = \Pr(s_{t+1} \mid s_1, a_1, \ldots, s_t, a_t). \tag{2.25}$$

$$R(s_t, a_t) = R(s_t, a_t \mid s_1, a_1, \ldots, s_{t-1}, a_{t-1}). \tag{2.26}$$

In short: we can predict the next state distribution and next reward from *just* the current state and action. This is incredibly helpful for simplifying analysis. Moreover, if any problem is *not* Markov, we can always just roll the last k steps of the world into the state representation, thereby yielding a Markov model.

The central operation of RL is the repeated interaction between an agent and an MDP. Critically, the interaction assumes that the agent knows everything about the current state of world: there is no notion of hidden information (aside from not knowing the causal rules or reward structure). An extension, the Partially Observable MDP (POMDP) [76], does not make this assumption. In either case, the agent interacts indefinitely with its world, trying to update its beliefs about what exists in the world and how to take actions to maximize reward.

The RL framework can also serve as a framework for understanding human behavior. Humans perceive and take action. While not altogether obvious that real valued rewards govern or incentivize action, recent neurological evidence suggests that reward prediction is a fundamental exercise of the human mind [131, 11, 115]. Additionally, any task with a goal can be redefined purely in terms of reward maximization, so goal directed agents are naturally captured, too. To see why, consider the following. Let the predicate G assert that some desired property of the world is satisfied by the current state of affairs. For instance, a person wants to satiate hunger or arrive at the airport on time. Such a goal can always be mapped to a reward generation, R_G, that provides an agent with 0 reward when the goal is not satisfied, and 1 when the goal is satisfied. Thus, the framework is surprisingly general.

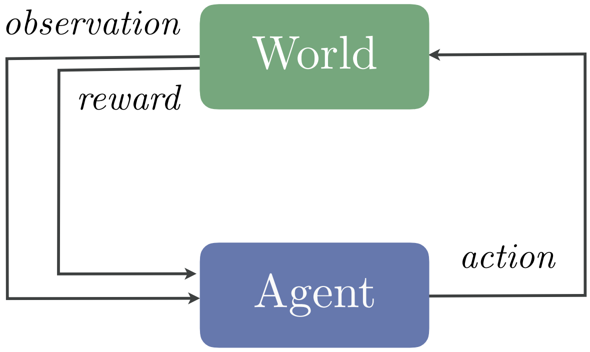


Figure 2.1: The reinforcement learning problem: An agent interacts with a world by taking actions and receiving (1) observations, and (2) rewards. The goal of the agent is to take actions that maximize long term reward.

Some quick clarifications. What is an observation? Intuitively, it's a single time slice of the input to sensory organs. For a robot, naturally, this input will instead be the camera (or other sensors') input for a particular time. What is a reward? Typically, a real number, denoting how desirable a particular configuration of the world. For simplicity, we assume rewards are real numbers in the interval: $[\textsc{RMin}, \textsc{RMax}]$, with RMin and RMax the minimum and maximum achievable rewards respectively. Lastly, what are actions? We suppose for simplicity that any agent has available to it some finite set of actions $\mathcal{A} = \{a_1, \ldots, a_n\}$. These actions correspond to the *primitive* actions of the agent. That is, for a person, they refer to the muscle controls used to move our bodies, as opposed to a high level action like "drive to work". We suppose that additional cognitive machinery is needed to induce a "high level" action such as "drive to work" from a set of primitive actions. Forming such high level actions is an active area of current research [85].

An Example

We now consider what is perhaps the most iconic problem in RL: a grid world, used by the AI textbook by Russell and Norvig [128]. The Russell and Norvig grid world is a discrete, 3×4 grid, in which each state corresponds to the agent inhabiting one of the grid cells. The agent's possible actions are $\{$`up, down, left, right`$\}$. After each action, the agent receives its (x, y) coordinate as an observation along with 0 reward (unless denoted otherwise). The grey cell is a wall, as are the "edges" of the world, which simply prevent movement in that direction. This world is of course extremely simple. However, this simplicity can count as a virtue in terms of understanding approaches to RL – we mostly know what the agent should do, so the world can offer insights into diagnosing various approaches. Of course, the true goal of RL is to reorient approaches to solving this grid problem at more realistic scenarios where we *don't* know the structure of optimal behavior.

Critically, the agent knows *nothing* about the world in its infancy – it doesn't know where the +1 is located, or the semantics of the action `up`. It has to learn how the world is laid out and where the reward is from interaction alone. This is an extreme position to take, but it is not necessary to the formalism. We can of course suppose that any agent we study is endowed with initial knowledge about the world (or inductive biases that lead it toward more parsimonious solutions).

In this setting, we ask: how can the agent learn to maximize reward in its environment from interaction alone? We seek a strategy that will ensure, after some amount of experience, that an agent finds the goal **quickly**, while avoiding the "-1" cell? Of course, we could write down the following recipe: `right, right, up, up, right`. However, this solution is tailored to this problem, and would fail outright in even slight changes to the problem such as moving the goal or wall. Similarly, we could have the agent always choose a random action – surely it would eventually find the +1. But, it would also come across the -1, and might take a long time to get to the +1.

What we would really like is to find a *general* purpose algorithm that will solve not just the above grid world, but any problem like it, and quickly. For instance, suppose the room were ten times larger, or the -1 moved, or the wall moved, or even the goal – in all of these cases, we'd like the same method to solve the problem. Understanding the nature of such algorithms is the goal of RL.

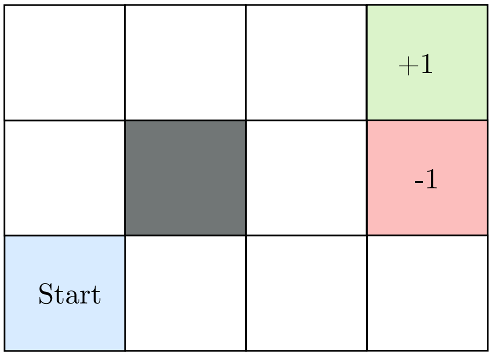
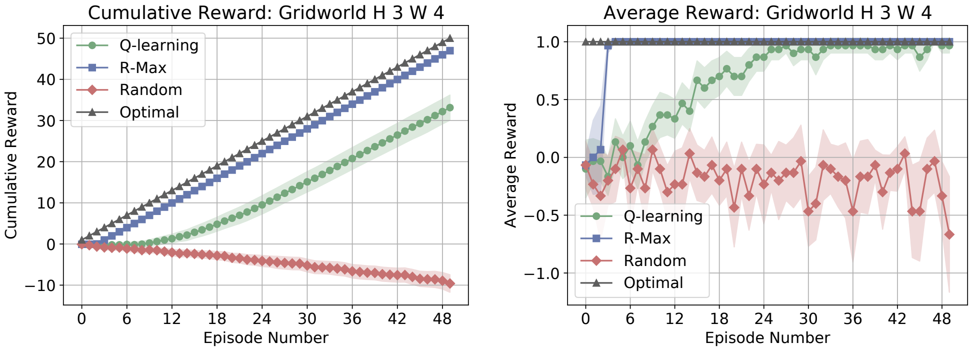


Figure 2.2: The classic grid world from Russell and Norvig [128]. The agent starts at the state (1,1), and moves around the environment in an attempt to reach the terminal state, (4,3).

As an illustrative example, I've included a plot showcasing the performance of several approaches to RL, experimenting with the above grid world, shown in Figure 2.3. In the experiment, each agent begins from a *tabula rasa* state – they know *nothing* about T or R – all they know is that there are 11 states and 4 actions. The agent gets to run for 20 steps, which constitutes one "episode". After each episode the agent is reset back to the beginning, so the best possible strategy would get +1 each episode. The results showcase the performance of three different strategies for learning to take actions: (1) Q-Learning [173], (2) R-Max [19], and (3) A random actor. Broadly, Q-Learning describes instinctive approaches to decision making, fitting into a category called "model-free" methods. The central guarantee of Q-Learning is that, in the limit of experience, Q-Learning will perform optimally. So, if run forever, it will eventually do better than R-Max. Conversely, R-Max explicitly constructs a model of the world to learn, and is thus more often compared to deliberative



(a) Cumulative Reward

(b) Average Reward

Figure 2.3: Two views of the same results from three simple agents (blue, green, and red) learning on the grid world task (higher is better), with black denoting optimal behavior. In both plots, the x-axis denotes *episodes*, which consists of fifteen steps of interaction. After those fifteen steps, the agent is moved back to the bottom left state, and the problem resets (but the agent gets to keep any knowledge acquired in the previous episodes). Roughly, the x-axis in both plots measures how many steps of interaction with the world the agent gets. On the left, the y-axis denotes total cumulative reward, while the y-axis on the right denotes average reward (for the same experiment).

models of planning and cognition (and so falls under the "model-based" methods). R-Max comes with a guarantee akin to the PAC property discussed in the previous section: the property of PAC-MDP [152]. The PAC-MDP property states that with high probability, after a specific number of experiences, the agent will achieve near-optimal behavior. So, in general, R-Max has much stronger guarantees – we know how it will do with finite data.

We find that very quickly, R-Max (pictured in blue) can find an action selection strategy that ensures it always reaches the goal. Note that after 50 episodes, the agent has received almost 50 reward. So, on almost all of its trials, it has found a path to the +1 while avoiding the −1. Note that it sits just below the black line, which denotes the reward received by the optimal policy. Conversely, Q-Learning (in green), takes its first 20-30 episodes to find a reasonable strategy. Once it has discovered that strategy, it is able to perform well in the final 20 or so episodes. Lastly, the random approach (in red), never fluctuates far from 0 total reward; sometimes it receives +1, others −1, so in expectation, it gets about 0 (though the −1 is slightly easier to stumble into).

One final critical note is that the notion of *expected utility* from decision theory has a natural analogue in RL, which we call *value*. Value denotes the *expected discounted utility* of a particular behavior. Using this quantity we can evaluate different behaviors and identify the best ones. Concretely, the value of a state, s, is defined as:

$$V^*(s) = \max_a \left(R(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s' \mid s,a) V^*(s') \right). \tag{2.27}$$

That is, value is the immediate reward an agent would receive ($R(s,a)$) plus the discounted expected future reward it would get – if it acted according to the best action a. Fortunately the above equation also comes along with an algorithm, called Value Iteration, that identifies the optimal behavior [12]. However, this assumes perfect knowledge of R and T, which, in the true RL setting, an agent has to learn from interaction (we don't grow up knowing how doors open, we have to learn this!).

2.5.1 Some Philosophical Issues

Naturally, the philosophical significance of RL will not be established by how algorithms perform on small, discrete, grid-based tasks. The essence of the model gives us a formalism for talking about what agents are capable of when they must both *learn* about their environment while trying to be rational in that environment. To access this theory, we will need to stick with the same core model, but will relax our assumptions in several critical ways. I will depart from what I take to be the three assumptions at the core of modeling the world as an MDP:

- Markovian physical laws.

- Full observability.

- Utility as reward maximization.

In our study of rationality, we will avoid making all three of these assumptions. Still, we will look to RL for insights by translating several of its foundational results from the utility maximization paradigm to rationality.

Markovian Physical Laws As discussed previously, the *Markov* property is said to obtain of a system just when the system's distribution of possible configuration at time x_t can be fully predicted just from x_{t-1}. It remains an open question as to whether there exist unchanging physical laws governing the world that depend only on local (in time or space) properties. If such laws were to exist, then it is likely that we can posit such laws have a Markovian form. Many microcosms that we encounter are surely Markovian, such as playing chess (and other board games), solving scheduling problems, aspects of dialogue, and so on. So, even if the full world lacks underlying Markov causal laws, we can gain insight into how far Markov models might take us in understanding behavior and rationality in a variety of situations.

Full Observability In traditional RL, the agent in question receives as input the *state* of the world at each time step. Critically, per the Markov assumption, the current state of the world is sufficient (along with an action) for predicting: 1) the next reward to be received and 2) the probability distribution of next states to be received. In the world, inevitably no agent can actually observe the full world state. Instead, we receive an ego-centric *observation* that is highly informative about our immediate surroundings, but does not tell us everything there is to know about the world. For instance, if we stand just outside a closed room, we don't know whether there is a tiger in the room or not. We just have to speculate based on our observations. We might lean close to the door and listen for tiger sounds, and update our belief about the current world state based on what we hear. Such a setup is captured by the Partially Observable MDP (POMDP) [76]. In a POMDP, agents explicitly received observations, which only give partial information about the true world state. The goal, however, is the same: take actions that maximize reward. For this reason, POMDPs are considerably more general: every MDP can be turned into a POMDP with the trivial observation function that where the observation received is exactly the current world state. Due to their generality, they are known to be difficult to solve. Further, much less is known about the nature of POMDPs, which leaves them as an open and interesting model to study in future work. For our purposes, we will exploit insights from RL primarily in the MDP setting, and effectively punt on the full POMDP problem.

Utility as Reward Maximization Traditional stories of rationality focus on one of two things: proper alignment of an agent's beliefs or credences to the facts of the matter, or making appropriate decisions in a decision theoretic setting. In RL, we find a treatment that is most closely aligned with the latter case of making effective decisions. As discussed previously, the ultimate measure of an agent's effectiveness is its capacity to maximize reward. Such a metric yields several specific methods for characterizing an agent's effectiveness. These include:

1. The sample complexity of exploration [79].

2. Regret bounds, as in [118].

3. A KWIK bound on the model of the MDP [94].

At a high level, each of these techniques characterizes how well an agent is said to perform, in general. Sample complexity, akin to computational complexity (and PAC bounds) asks how many *experiences* are needed by an agent before it is guaranteed to make good decisions. This is roughly like saying: how much evidence do you need about your world until you're guaranteed to have a good enough model of it to make good decisions? Regret measures the magnitude of your mistakes as you learn. Just like computational complexity, both regret bounds and sample complexity bounds tend to come in asymptotic form.

It is precisely these forms of bounds, though, that will be incredibly important to our unfolding of bounded rationality through RL. Consider the following example, a k-armed bandit:

Example 6. *Consider a k-armed bandit [53], in which you play a game that proceeds in T rounds $1, 2, \ldots, T$.*

Each round you must pull one of k possible slot machine arms. Playing arm i collects a sample from some unknown payoff distribution R_i.

The goal of the game is to pull arms $i_1, i_2, \ldots, i_T$ that maximize the received payoff.[4]

The difficulty of the game is the so called *exploration-exploitation dilemma*: after t pulls, you've collected some amount of information about a few slot machines. You have two choices:

1. Do you take advantage of the knowledge you have about the arms you've already tried and keep pulling them?

2. Or, do you try new arms, running the risk that they won't be as good as the one's you've tried already?

The first response is the decision to *exploit* the knowledge you already have, while the second is the choice to *explore* the world more to learn about other options.

[4] Other criteria are often of interest too, such as identifying the arm with the maximal expected value.

The explore-exploit dilemma introduced in the previous example shows up all over the place in our lives: suppose you go to a restaurant – do you try new food or order something you know you like? Do you move to a new city you've never been to or live in a place you know you are comfortable? Do you tackle new philosophical territory or do you keep pushing the areas you know you'll make progress in?

Such a dilemma is omnipresent. Indeed, in any decision making setting in which the agent lacks perfect knowledge about the world, the agent must confront the exploration dilemma. This is a key factor of RL that is not present in any existing theory of rationality. We ask: do agents *explore* rationally? In a decision theoretic setting, a good exploration strategy is essential to behaving rationally (with respect to some utility function).

So, while the full RL framework makes three additional assumptions that are philosophically inadqeuate, we find that 1) the framework is still immensely useful, even with those assumptions, and 2) we have a clear path toward relaxing these assumptions. Indeed, further work is likely needed to further expand on these routes. Here, we focus on what we can get away with in our understanding of rationality from the formal tools we have introduced thus far.

To summarize what we've introduced, we now have tools for analyzing:

- The number of *computations* needed to solve certain problems in a variety of ways (worst case, average case, approximation).
- The amount of evidence needed to learn a concept (under the PAC setting).
- The number of experiences needed to effectively make good decisions (from RL).

Collectively, each of these points will let us clarify the picture of rationality further: agents must not only solve problems given a reasonable computational budget, but are expected to learn good concepts, and learn about their environment given finite experience and evidence – we can now measure how well, in principle, an agent can do, subject to constraints on all of these components.

Chapter 3

Bounded Rationality from Reinforcement Learning

We now flesh out rationality through RL. In order to gain a foothold into mathematically quantifying an agent's capacity for the so called rationality required projects [41] including concept formation, planning, and exploration. Using RL, we can inspect how different agents can learn, explore, communicate, plan, and update their beliefs. Arguments for similar formalisms have been raised by Ortega and Braun [116], Harb et al. [59], Zilberstein [177, 178] – see their work for further support and background. For a recent summary of algorithmic perspectives on human decision making, see Christian and Griffiths [29].

3.1 Physical Agents, Physical Limitations

Consider any physically instantiated agent. By definition of "physically instantiated", the agent must have physical form in some world it inhabits, and by "agency", the agent must make decisions by selecting from a collection of competing options. To be most useful in our study of rationality, the properties that limit our agent-space must be those that necessarily obtain of all agents of relevance, which we take to be: people, animals, and artificially intelligent agents such as robots.

In the extreme, consider the omniscient physical agent, which has all knowledge about the world it inhabits. This is perhaps the most idealized a physical agent can be. How are we to restrict this agent? The usual story of the ideal rational agent adds constraints of the kind discussed in the previous chapter: closure under deduction, probabilistic coherence, and so on. Bounded rationality suggests further imposing a mixture of constraints in the form of thinking time and space (perhaps via complexity theory) and and lack of world-knowledge (perhaps via learning theory). RL serves as a unifying formal language for describing the interplay of these considerations for physical agents.

We suppose the following properties must obtain of a physical agent:

Definition 6 (Physical Agent): *A physical agent is any agent (an entity that chooses among competing outcomes) that satisfies the following properties:*

- *Has finite computational constraints (both time and memory).*
- *Receives ego-centric, bandwidth-limited percepts of the world.*

The above definition is not intended to be controversial—indeed, it is only given in order to sharpen our inquiry. Moreover, it is based, in large part, on the original characterizations of boundedly rational agents given by Simon, and of computationally rational agents by Gershman, Horvitz, and Tenenbaum [49]. I defend each property in turn.

Finite computational constraints (in the form of time and memory) also come across as natural: per the arguments raised previously, it is unhelpful to consider what an unbounded agent will do. It is akin to considering the agent that can solve some arbitrarily difficult problem, perhaps even The

Halting Problem. Given an oracle to the Halting Problem, many other undecidable problems can we solved. In what sense is this consequence meaningful for understanding our own decision making? It gets us no closer to understanding how we *should* be updating our beliefs or making decisions. The problem is *how* to apply such constraints. As we explored in the previous section, some mixture of computational complexity analysis seems appropriate, but there are many competing theories, and it is not obvious how to choose between them.

Ego-Centric, Bandwidth-Limited Percepts is intended to force the agents we study to receive evidence in a form that is compatible with our own sensory tools. Why? Well, again, if our interest in rationality comes from trying to either give guidance on how to be rational or place blame for invoking irrational belief forming methods, then it is unhelpful to do so from the perspective of an entity that so blatantly violates the constraints we operate under. If rationality is about placing blame – again, how are we to get a clear sense of what behavior is blameworthy if we don't have a (remotely) accurate model for describing rationality? As a consequence of this property, it is implicit that our idealized rational agent must enjoy some form of physical existence, too. Then, through limited-bandwidth perceptual tools, the agent can gather evidence about its current world, *but must do so by acting and observing its current surroundings.* This is the critical departure. As one might expect, the tools we built up in the background chapter will be immensely useful for formalizing and studying this constraints. The PAC framework already told us how to relate the amount of data received in a simplified setting to the generalizability of the concepts the agent can learn. The PAC setting, though, is too simple – we really need an agent to perceive evidence as it wanders. This is where RL will enter the picture.

The above two properties define what is meant by "physical constraints"; any agent that aims to be rational must confront the fact that it has limited resources to reason with, and has limited sensory capacity to obtain evidence from its surroundings, by necessity. Other properties have been proposed, such as *anytime* decision making [177], in which an agent must be ready to make a (pretty good) decision at any time, even if it hasn't yet finished any relevant calculations. Another fruitful route has been to investigate *metareasoning* [62, 59, 178, 56], in which agents reason explicitly about the cost of performing more computation. These considerations are extremely important, but beyond the scope of our present discussion.

Let us begin with an example.

Example 7. *As in the k-armed bandit problem, suppose an agent is presented with k boxes of toys. The agent is asked whether there is a pink toy elephant in one of the boxes.*

The agent is thus responsible for searching through the boxes to determine if the pink elephant is present, and if so, which box it's in.

Supposing there is exactly either one pink elephant or zero, there are precisely $k+1$ competing hypotheses to choose from. So, from the PAC result presented by Theorem 2.18, if we can change the problem to ensure we collect evidence as in the PAC setting, we know that we will need at most:

$$\left\lceil \frac{\log \frac{|\mathcal{H}|}{\delta}}{\varepsilon} \right\rceil, \tag{3.1}$$

samples from our boxes to determine whether the elephant is in a box with probability $1-\delta$. Here, $\mathcal{H}$ is the size of the chosen hypothesis space, $\varepsilon \in (0,1)$ is an approximation parameter, and $\delta \in (0,1)$ is a confidence parameter.

However, our evidence is no longer independent and identically distributed. We have to actually choose which boxes to search in, and how to search them; this is a necessary consequence of being a physical agent. So, the optimal strategy here, and even the worst case bound, is dramatically altered from the case where our evidence is just presented to us in a straightforward way. This is precisely why the ego-centricity is so important: our agent actually has to *choose* which evidence is going to be most informative for updating beliefs. We must choose which evidence to collect in a rational manner – given limited resources, our evidence collection process will dictate 1) how well our beliefs match reality, and 2) how effectively we can make decisions (suppose we were granted higher utility upon finding the elephant sooner). Our agent must actually play out the experience of looking through box 1 for the elephant, then looking through box 2, and so on. Depending on how much time the agent invested into box 1, it may again be rational the agent to revisit the box, given memory constraints ("did I check box 1? Did I check it thoroughly enough?"). Moreover, as the agent digs around in box 5, they may learn something about the contents of the other boxes: there can be pink tigers! Or, perhaps there can be blue and red elephants, and they're the size of a peanut. This may change the agent's strategy for choosing to go back to box 1 or not.

In the above example, we saw how an agent that receives data in an ego-centric way is faced with a challenging belief formation problem. What would the ideally rational agent do? One answer would be that it follows the ideal exploration strategy: in bandits, such a strategy is known [53]. More generally however, we don't yet know what such a strategy would look like. It is unclear: the theory simply lacks the appropriate machinery to discuss hypothesis testing of this form. Different evidence collection strategies will lead to different beliefs, given a resource budget: once you see that the first box is *all* blue animals, containing no elephants, do you suppose each box is color coordinated? Or animal coordinated? Suppose you find a toy elephant in box 3 the size of a thimble – would this suggest that there may have been an elephant, smaller than what you originally believed, in a box you already searched? This kind of belief updating is not well modeled by other theories.

RL is perfectly suited to study such problems. The k-armed bandit codifies a simplified version of the above problem, which has led to a great deal of analysis and fruitful study [30, 53]. From the bandit literature, we can state that, in a simplified version of the pink elephant problem, *any* agent will need a certain number of rounds of looking through boxes in order to find the elephant. We can define precisely the computational and statistical resources required to solve the elephant problem, in general.

Suppose the other constraint now enters the picture: in addition to having to *choose* which evidence to process, agents must also do so given only finite resources (in terms of computational cycles and memory). As explored in Chapter 2, we have a variety of tools for describing such constraints, though none are quite perfect. As we seek to characterize the ideal physical agent, we will jointly limit its resources in addition to forcing its evidence to be perceived in an ego-centric, bandwidth limited way, as discussed in the pink elephant example.

3.1.1 Characterizing the Ideal Practical Reasoner

So far we have clarified "realistic physical limitations", and still require further clarity about "capacity to reason" and "idealized".

Capacity to reason translates to an agent's ability to make effective decisions. In the utility maximization sense, an agent is said to reason well just when it does a good job at maximizing utility. The major departure we have made so far is that the agent is uncertain about the nature of the world: it doesn't know the causal rules of the world, or the appropriate labels for different objects. It must learn these facts during the course of its existence.

Our previous discussion about ego-centricity suggests the agent *doesn't* necessarily know what is either feasible or desirable. This is precisely the formation offered by Reinforcement Learning: the agent must learn about both what is feasible and what is desirable, all the while making decisions to maximize utility. Agents must choose how to gain evidence so as to ensure they're maximizing their capacity to learn about their world. Thus, when we talk about an agent's capacity to reason in RL, we will typically talk about utility maximization. Fortunately, utility plays an explicit and important role in the MDP formalism – an agent's expected discounted reward is defined under a fixed decision making strategy, $\pi : \mathcal{S} \rightarrow \mathcal{A}$, according to the Bellman Equation [12]:

$$V^{\pi}(s) = \underbrace{R(s, \pi(s))}_{\text{Immediate Reward}} + \overbrace{\gamma \sum_{s' \in \mathcal{S}} T(s' \mid s, \pi(s)) V^{\pi}(s')}^{\text{Discounted Expected Future Reward}} . \tag{3.2}$$

Here, V denote the *value* of a state under a policy, π, which indicates a behavioral strategy for every state (that is, $\pi(s)$ outputs an action). $R(s, \pi(s))$ is the immeidate reward received by taking the action $\pi(s)$ in state s, $\gamma \in [0, 1)$ is a discount factor expressing an agent's preference for immediate rewards vs future rewards (the agent prefers immediate rewards as γ gets closer to 0), and $T(s' \mid s, \pi(s))$ is the state transition function: when the agent follows π in state s, what world state are they likely to end up in next?

This equation yields a simple recursive mechanism for computing the expected utility an agent will receive for a given problem. Note that in order to compute the above function, we need to know 1) the behavior to be evaluated, π, 2) the true transition function describing the causal rules of the MDP (T), and 3) the reward function (R). If an agent knew these quantities, it could compute the value of its current policy, and thereby improve it by searching for a better policy. The central assumption, however, is that the agent doesn't know R or T. Instead, the agent must determine what is feasible (learning the causal rules of the world, T), and what is desirable (learning the utility-filled states of the world, R), by wandering through the world and experimenting.

The job of any RL agent is to learn about R, T, and V as best as possible, so as to come up with a decision making strategy π that maximizes V^{π} in the states of relevance. This is precisely how we will measure an agent's capacity to reason.

To explore this point further, let us consider an extension of the pink elephant example:

Example 8. *Now, instead of being presented k boxes, our agent is placed into a grid world, similar to the Russell-Norvig grid from Figure 2.2. Consider the slightly more complicated grid pictured in Figure 3.1. The agent is pictured as the blue triangle in the bottom left of the world, and is searching for the star. The agent's only sensor can determine whether the agent is currently in the same location as the star or not, and can determine the name of the state it is currently in (that is, it can learn whether it has been in the state before or not).*

At the agent's birth, we suppose it does not know about the structure of the grid. It must learn about the structure by wandering around and learning the effects of its actions, similar to how we learn to control our fingers to grasp complex objects [145].

What would the ideally rational agent do? Suppose the agent's mission was to form perfect beliefs about 1) how the world worked (what are the consequences of its actions in each cell?), and to 2) find the location of the star.

As in the case of the pink elephant, the core challenge is that the agent *must* choose wisely as to how it learns about its world. One option would be to act randomly for N action choices, then use that information to learn. Based on the set of N experiences, an agent might get lucky and find the star, or might learn that executing `right` tends to have a certain kind of effect. Alternatively, an agent that decided to try `left` for all N actions would just run into the wall repeatedly and learn very little about the world. So, different strategies for collecting evidence yield different possibilities.

As we saw in the first grid example the approach called R-Max did considerably better than Q-Learning. The Q-Learning algorithm roughly tries to explore randomly, which leads to considerably worse behavior. Conversely, R-Max picks which hypotheses to test efficiently, and can consequently learn to make better decisions more accurately. At a high level, R-Max always assumes it is in a world where there is some extremely high reward in every single state in the world. In this way, it will always try to plan to get to states it has not yet seen (unless it can actually get a lot of reward in its immediate vicinity), thereby encouraging exploration.

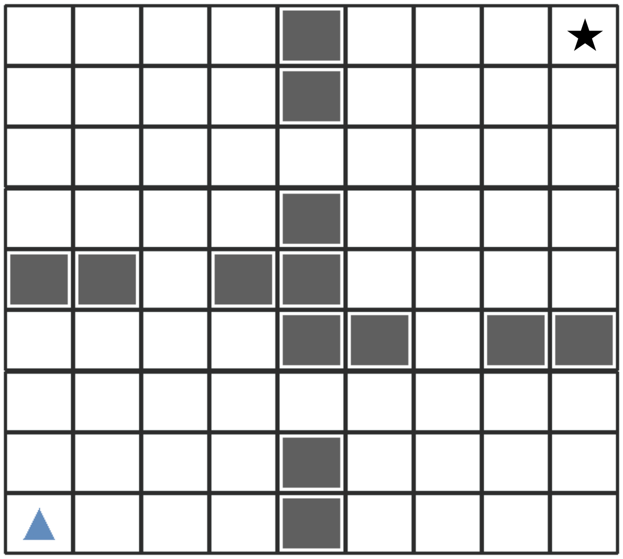


Figure 3.1: The Four Rooms grid world domain from [156]. The agent starts at the triangle and receives +5 reward when it finds the star.

In the above example, each agent is responsible for both gathering evidence *and* making best use of the available evidence to make good decisions. A good decision, with consideration of the exploration problem, can be one that either *explores* effectively (so as to gather the right information for later us) or one that *exploits* effectively (so as to take advantage of the information already gathered). Further, any rational agent must make an appropriate trade-off in how much it explores or exploits. An agent that over-explores will waste valuable time not maximizing utility, and will thus under perform. An agent that under-explores will settle with a decision strategy that is sub-optimal relative to some as-of-yet undiscovered solution. This problem is core to any pink-elephant like scenario.

Fortunately, the RL literature has developed concrete measures for evaluating the capacity of an agent to effectively explore its environment. There are roughly two such measures, but both target the same point:

1. Regret: The regret of an agent measures how much worse the agent is relative to the optimal strategy (taking full advantage of perfect world-knowledge). So, for instance, if the optimal strategy (assuming full knowledge of the world from the first time-step) yields a score of +5 in the grid example, then the regret of an agent is now much less than +5 the agent receives after taking into account its exploration time. As with computational complexity, the focus is again on orders of magnitude based on the problem size. In decision making, our problem size can be cast in terms of quantities like the *horizon* of the problem, H (the number of decision steps the agent can act for), and the *size* of the problem, measured in terms of the size of the state-action space $|\mathcal{S}| \times |\mathcal{A}|$. The lower bound on regret is known to be $\Omega(\sqrt{H|\mathcal{S}||\mathcal{A}|N})$ [119], with N the number of time steps.

2. Sample complexity of RL: The sample complexity, like computational complexity, measures the number of samples an agent need before it can act near-optimally with high probability [79]. A long line of work has explored increasingly better sample bounds [80, 19, 157, 151, 152]. The Knows What It Knows (KWIK) framework extends sample complexity to capture a more challenging learning objective that is well suited to RL, too [94].

In either case, we gain, as with computational complexity, a formal measure for evaluating an agents decision making strategy under realistic assumptions. It is important to note, however, that both measures are limited to discrete and finite Markov models (MDPs, discussed previously). Recent work has extended sample complexity to a more general class of decision problems [70], suggesting that there are pathways to realizing similar bounds in the general sense.

3.2 Putting It All Together

Suppose we want to inspect a resource bounded agent $\mathscr{A}$. Really, we want impose constraints of a variety of kinds. The upshot of the theory we have introduced is that we have readily available methods to define each of the above constraints:

- The agent can think for no more than N "steps" per decision.

  → Impose a computational complexity restriction on the agent's available computing time, per time-step.

- The agent can use no more than M bits for its memory at any given time.

  → Impose a space complexity restriction on the agent's available computing memory, per time-step.

- The agent has only partial knowledge of its world, as represented by its initial beliefs over the world (MDP) it inhabits.

  → Suppose the agent has a prior on quantities of relevance, like the transition model of the world (T) or the value/utility of a state (V). Or, even simpler, we can gift the agent with partially accurate initial models, such as $\hat{T}$.

In the end, questions about *practical* rationality are posed about a particular agent's capacity to solve different problems, subject to these constraints. I take the resulting account to serve as a unified picture of resource-bounded practical rationality.

Chapter 4

Concept Formation and Rationality

The main claim defended in this chapter is that bounded rationality is not only about elegant symbol crunching, but also about choosing the right symbols so that symbol crunching is *easy* and *effective*. By "symbol", I will roughly mean "concept".

What is a concept? I adopt Susan Carey's view [22]: concepts are the atoms of thought. They can make up beliefs, thoughts, and consist in mental representations. Beyond these large strokes, I will remain agnostic to the precise character of concepts. If we take some view in the neighborhood of Fodor's Language of Thought hypothesis to be correct [44], then a concept is just an atom of the language of thought. These might be the possible categories that we use to update and generalize beliefs (such as "dog", "sunny", or "Tuesday") or the properties we used to describe different experiences or objects ("large", "fun"). All that is really needed is that concepts factor into our ways of breaking down and thinking about the world.

I will primarily focus on two special kinds of concepts: *world-state* concepts, that characterize our surroundings (past, present, and future), and *ego-centric behavioral* concepts, that describe what kinds of behaviors an individual agents is capable of.

For example, suppose you find yourself in a kitchen baking a delicious loaf of bread. Relevant state concepts might consist in the way that you break down and organizes aspects of the kitchen: cupboard, utensils, measurements, dish, surface, knife, and so on. They likely also describe the current progress made on the dish; has the dough risen enough? is the oven heating? is the dough too gooey? Action concepts describe the high level behaviors like "make the dough", but also lower level behaviors like "measure $\frac{1}{4}$ teaspoon of vanilla extract". In this example, one might see how the *wrong* choice of concepts (of either kind) can prohibit effective baking. Going forward I will call this first class of concepts *state* concepts, and the latter *action* concepts.

To leverage mathematical insights from learning theory, it will also be convenient if our treatment of concept tracks (at least partially) the boolean functions learned in the PAC learning framework described in Chapter 2: a function, c, that assigns entities in our domain into two groups (the "has the property" and the "do not have the property"). Really, then, the boolean functions learned in PAC learning are *extensions*: they divide a given domain into two sets of things. Of course the concepts actually at play in human cognition are considerably more complex than these functions, but it will be important to preserve some tracking between the two. For example, we make use of "dough" as a concept that differentiates some objects from others. This can be spelled out in terms of cognition, the phenomenology of experiencing kneading dough, and in terms of a function c that labels images as containing dough or not containing dough. All of these approximate what is meant by concept. It is both slightly remarkable and unsurprising that our culture, language, and evolutionary lineage have all converged on concepts that support communication of our experience, recollection of memories, accurate symbolic reasoning like complex math and predictions about our future, as well as a fluid perception-control loop capable of supporting our many physical feats, such as baking.

Carballo [21] studies the utility of concepts for *epistemic* rationality, as opposed to practical rationality, though many of the considerations are similar: Carballo's main argument is that concepts can do a better or worse job of naturally carving the world at its joints. I view our theories as compatible: in practical rationality, a concept X is better than Y just if an agent with resource constraints can make better decisions if thinking in terms of X as opposed to Y. The crux of this argument says that without good concepts, resource bounded agents can't plan well, or can't generalize well (concepts need to describe not just our experiences now, but ones we have yet to encounter, too). Prior psychological studies suggest that the primary role of concepts is to support good categorization [54, 126, 33, 112] and generalization [25], so much of what I argue here is not novel. Further philosophical work explores the use of concepts, or their status in epistemology more generally, such as Gardernfor's conceptual spaces account [47], which also bears connection to AI through induction [46]. Further work in AI has explored bounded rationality [176, 63], exploring planning under resource constraints [14], metareasoning [64, 178, 56], or on construction of *anytime*[1] algorithms as a criteria for operational rationality [177]. The primary novelty in this work is unifying aspects of these accounts (psychological, AI, and philosophical) through the role concepts play in boundedly rational behavior.

4.1 Baking Concepts

Let us again turn to the baking example. This time, suppose Matt, a person who knows nothing about kitchens, baking, or any other relevant concepts, is presented with a standard kitchen for the first time. What might Matt need to know in order to bake a loaf of bread? Consider first

[1] An anytime algorithm is one that can always output a solution, even if the algorithm has not yet finished its execution. Such a criteria is likely very important for practical models of behavior, as agents should always be able to come up with a solution, even if its not the best one

the rather extreme RL position; Matt knows nothing about the world and must learn by randomly executing his most basic actions.[2] Matt flails his arms around arbitrarily hoping to receive some positive reward (which, in this case, is provided through machinery in his brain). Perhaps, given enough time, bread will be made. However, this is clearly an undesirable path toward making bread; it is just as likely the kitchen will catch on fire or Matt will injure himself. What, then, can we endow Matt with in order to speed the process, and make him a more capable baker? I take the natural response to be: an appropriately useful model of the contents and causal rules of the kitchen. Matt needs to know: What is flour? What is water? Where does water come from in the kitchen? What is an oven? What is temperature? These (and others) are roughly the state concepts needed for baking.

Many of these concepts are *necessary* in order to even get off the ground with baking. But, I here aim for a stronger point still: there is additional gain to be made by honing concepts further beyond this initial set. There are two ways in which state concepts can be improved in order to make decision making more effect: state concepts can *generalize* to enhance learnability in unseen situations, and state concepts that make *planning easier*, by allowing agents to reason with a more compact model of relevant states of affairs.

Consider the fact that even an experienced baker will never have baked under precisely the same conditions twice – ingredients and dates change, hardware erodes, and so on (perhaps the baker finds themselves in a new kitchen!). We thus require that these state concepts are sufficient for translating across different experiences along many different dimensions. Even if they don't afford instantaneous perfect decision making, they do need to support the flexibility we expect of a competent baker (and decision making more generally). If a master baker were to look at a bowl

[2] I will occasionally use the term "basic" or "primitive" action to denote the lowest level of muscle/motor control actions available to an agent, like shifting joints/fingers.

and say, "Ah, I've never seen a bowl of *that* diameter! I can't bake with this", we would likely be reluctant to call them a master (or perhaps call them a bit too picky in their choice of bowls). It is hard to even take seriously the notion that a baker wouldn't know *how* to use a new bowl. In this sense, we expect our state concepts to will generalize well across the experiences we encounter. We require that we can react to dynamic and as-of-yet unseen phenomena. This is the first sense in which state concepts can be improved over their usual bare necessities.

Second, state concepts can afford quick planning. Planning roughly captures the process of deliberating over sequences of possible behaviors and choosing the most desirable of such possible sequences. The right state concepts can ensure that each step of the plan makes substantive progress toward overall progress; planning at the level of "what step of the recipe am I at?", as opposed to, "how many more muscle twitches do I need to execute in order to knead the dough?" can ensure faster planning. In this sense, it is hard to imagine planning without *also* considering an agent's action concepts too – plans deal explicitly in action. The idea is roughly the same. With baking, thinking through the sequences of actions Matt might take in order to make the dough, for instance, is easier if the right state-action concepts are chosen. Faster planning means that any agent with resource constraints can actually do *more* planning with the same resources. And, more planning (usually) means better decisions.

As suggested, the same reasoning that underlies the need for good state concepts (better generalization, faster planning) supports the need for good action concepts. Now, let us dive in more deeply.

4.2 A Closer Look at Good State-Action Concepts

We now turn to a more concrete example than baking to showcase precisely what we mean by generalizability and fast planning.

Consider the simple navigation tasks pictured in Figure 4.1. On the left we see two original problems: the agent must learn enough about its world so as to navigate to the top row. In the bottom example, the agent must also reason to find its way through the narrow doorway shown in blue. Each of the two problems are modeled as Markov Decision Processes (MDPs): the circles in the image represent states of the world, and the edges represent the consequences of the agent's actions in each state. For a recap of MDPs, see Chapter 2.5.

In this example, concepts are how the *agent* chooses to distinguish between states of the world and its own actions. In the RL literature, this is typically called *state abstraction* [38, 5, 73, 93]. On the right two figures, we find a change of state representation (Figure 5.1b) that reduces the number of states in the world, and a change of behavior representation that adds new long horizon actions to the agents model of the problem (Figure 4.1d). The idea of the example is this: imagine yourself in a long hallway. Spending precious mental resources on parsing every tiny change in the underlying world as a novel state will consume resources better left for other processing. Thus, making use of a compact state representation that still tracks with the actual goings on of the world leaves agents more free to perform more given their budgets. It is in this sense that choosing the right concepts underlies rational behavior. If an agent did *not* choose the appropriate state or behavior representation, then the agent's decision making will be worse than it could be, even with the same budget. If an agent can decompose its decision problems in the right way, problem solving becomes easier, so it can squeeze more value out of each mental calculation used to solve a problem. Of course, I haven't yet stated *why* decision making gets better under these concepts.

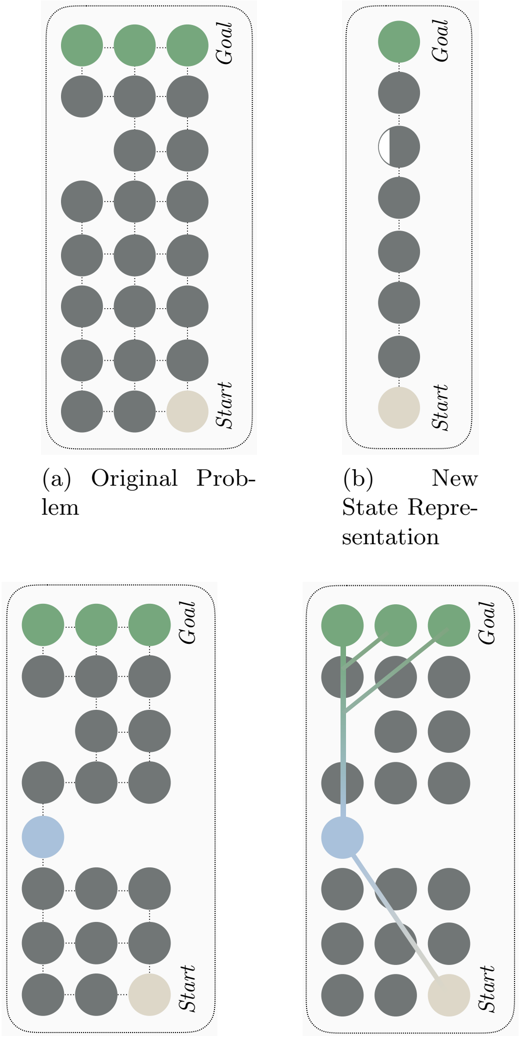
We discuss this point now.

Let us begin by returning to Simon's (and Von Neumman/Morgenstern's) chess example. Suppose a chess playing agent can only make a number of queries to the tree of possible games in some initial "training" phase, or perhaps has had the pleasure of playing some number of games, N, and must use these experiences to guide its play in some new $N+1$st game.

Now, this agent finds itself in a new board configuration it has never before seen during any of its prior games. How should the agent respond? One naive route is to charge ahead and assume that all new outcomes may be equally fruitful, so every move should be treated equivalently until more is known about them. However, surely this route does not take full advantage of the evidence gathered from prior games. In the same way that a chef in a new kitchen can transfer their beliefs and insights from prior experiences, we expect a chess player to do the same.

How, then, should the agent take into account knowledge about different board configurations and different board games, now that it finds itself in a new board layout? This is precisely the classical problem of generalization studied by supervised learning, as in learning theory. Moreover, this is effectively the classical problem of induction [65]. How are agents supposed to apply knowledge from past experiences to a new, similar experience?

The psychologist Roger Shepard provides one answer [135] which he coins the "Universal Law of Generalization". A critical aspect of this law is the notion of *psychological space*, which defines a landscape of entities that a given reasoner holds in its psychological focus [134**?**]. This space is intended to capture how we as agents break apart our experiences and stimuli – a beach ball is likely to be closer to a volleyball in psychological space than a beach ball is to a tomato (unless, perhaps, the beach ball is bright red). According to Shepard's view, our ability to generalize across similar experiences can be explained based on an inverse power law of the distance between experiences



(a) Original Problem

(b) New State Representation

(c) Original Problem

(d) New Behavior Representation

Figure 4.1: Two different navigation tasks (top and bottom) with different state (top right) and action (bottom right) concepts that can significantly decrease the complexity of the problem.

in psychological space. Recent work by Sims [139, 140] extends Shepards program via information theory, giving a similar perspective to the one we advocate for in the next chapter.

Returning to our chess example, Shepard would suggest that, if we had already pre-determined the appropriate psychological space for reasoning about board layouts, then we would be able to apply our knowledge of *similar* board layouts to this one. In essence: if we found ourselves in a similar position before, and have knowledge of what constitutes a good move in this prior experience, we can export our knowledge directly to our new found layout. This is rather intuitive, but is missing several critical pieces, the most glaring of which is *where psychological space comes from.* If we can, however, get our hands on the right notion of space, here, then we can generalize effectively.

4.2.1 Concepts and Generalization

Surely there exists some decomposition of chess board space such that every board with optimal move a_i is grouped near one another. In fact, we can prove this:

Lemma 1. *There exists a representation of chess boards such that boards that are said to be close to one another are guaranteed to have the same optimal move.*

More formally, let $\mathcal{S}$ denote the space of possible chess board layouts. Let $\mathcal{A}$ denote the possible moves for a particular agent. Let $d : \mathcal{S} \times \mathcal{S} \rightarrow [0, 1]$ define a measure on state pairs. Then, there exists a function, $\phi : \mathcal{S} \rightarrow \mathcal{S}_\phi$, that induces a new state space $\mathcal{S}_\phi$ (the "psychological space") such that:

$$\forall_{s \in \mathcal{S}} : \min_{s' : s' \neq s} d(\phi(s), \phi(s')) = 0 \ AND \ \pi^*(\phi(s)) = \pi^*(\phi(s')). \tag{4.1}$$

Proof. Consider the π^*-irrelevance class of functions, ϕ, from Li et al. [93], which we here denote Φ_{π^*}. That is, for any $\phi \in \Phi_{\pi^*}$, for any two board layouts s_1 and s_2:

$$\phi(s_1) = \phi(s_2) \implies \pi^*(s_1) = \pi^*(s_2). \tag{4.2}$$

From the Bellman Equation, we know that there exists at least one optimal behavioral policy $\pi^* \in \Pi$. That is, π^* is a deterministic mapping from a given state $s \in \mathcal{S}$ to a single action $a \in \mathcal{A}$.

Thus, consider the *maximally compressing* element ϕ^*, that groups all possible states together such that the antecedent from Equation 4.2 is true:

$$\phi^*(s_1) = \phi^*(s_2) \Longleftrightarrow \pi^*(s_1) = \pi^*(s_2). \tag{4.3}$$

Now consider the trivial distance function, defined over the "psychological" state space, for any function ϕ:

$$d(\phi(s_1), \phi(s_2)) := \begin{cases} 0 & \phi(s_1) = \phi(s_2), \\ 1 & \text{otherwise.} \end{cases} \tag{4.4}$$

Thus, the function d paired with the function ϕ^* ensures that, for any two board layouts, s_1 and s_2, their psychological distance is 0 just if their optimal action is the same. This concludes the proof. □

Intuitively, the above lemma just says that we can always group our chess boards into different regions of mental space based on what their optimal action is. In essence this is the π^*-irrelevance abstraction from Li et al. [93]. Using this abstraction, when we go to evaluate what to do in a particular region, we just have to check what we did in similar (or identical) states.

Naturally this is problematic, as having such a function is a sufficient condition for completely solving chess. Still, the takeaway is this: for one such kind of decision making, the "psychological space" that is chosen to think about the particular problem can determine whether a strategy for generalization is effective. In some actual cases, like chess, there are better and worse representations that are in fact feasible to latch on to; this is in large part the aim of *representation learning* [13] that constitutes deep learning [90].

For example, using learning theory, we can rigorously establish a representation's capacity to *generalize* over a domain. The same general claim holds in the case of planning: certain concepts are better for ensuring fast, near-optimal planning.

4.2.2 Concepts and Planning

Planning, at its core, defines the problem of deliberating over possible sequences of behaviors to come up with an appropriate action to take. The problem is of interest in both the psychology community [110] and to artificial intelligence [114, 61, 86, 88, 89].

In general the planning problem is as follows: suppose an agent must choose action a_1 or a_2. As in the case of Matt baking in the kitchen, decisions have not just immediate consequences but long term consequences, too. Often the objective is to determine whether the utility, $U : \mathcal{A} \rightarrow \mathbb{R}$, of action a_1 is higher than that of a_2, denoted $U(a_1) > U(a_2)$ or $U(a_1) \leq U(a_2)$. One popular approach involves *Monte Carlo* approximation: suppose we are given a model $\hat{T}$ that can accurately predict the next state of the world s_{t+1} given the previous state s_t and action a, denoted $\hat{T}(s_{t+1} \mid s_t, a)$. In a game of chess, the rules of the game define the true model T, and any agent will approximate it. In a game like chess where all rules are known, usually $\hat{T} = T$. Assuming our agent has an accurate forward predictive model ($\hat{T} \approx T$) Monte Carlo approximation means roughly the following: simulate choosing some action a_1, then simulate acting randomly (or behave according to a plan) for some amount of time. How good is the overall outcome? Then, repeat this simulated experiment some huge number of times for each action a_1 and a_2. If repeated enough, our agent can get a reasonable estimate of the overall long term utility of each action, and use these estimates to determine whether $U(a_1) \geq U(a_2)$ or $U(a_1) < U(a_2)$. Indeed, such methods served as the core of the recent Go AI that achieved world champion level play , AlphaGo [136], and its predecessor, AlphaZero [137].

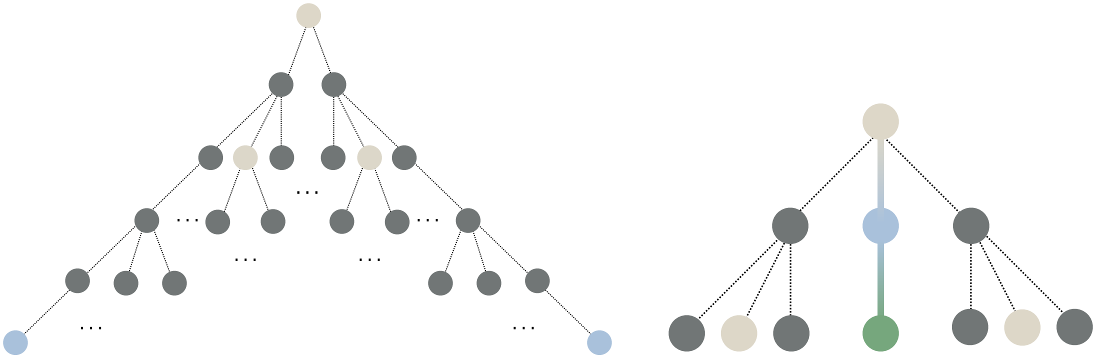
Planning is known to take serious computational work. The most relevant analysis of the difficulty of planning treats it as part of the same MDP formalism as RL; in this case, planning is known to be roughly $O(|S|^2|A||T|)$ in the worst case, with $|S|$ the number of states in the world, and $|A|$ the number of actions in the worst case, and $|T|$ is the number of bits needed to write down the environment's true forward model[97]. So, in the worst case, an agent needs this much computing time in order to come up with a good plan. Returning to our discussion of computational complexity, agents almost certainly don't need to worry about the full blown worst case analysis; the shortcomings of worst-case analysis from Chapter 2 are particularly poignant here. Solving the full planning problem would mean determining perfectly optimal behavior in every state of the world, even those that are unlikely to ever be reached.

Instead, we can consider the more practical question: if an agent has N seconds (or some limit on memory) to come up with a plan, how good can the plan be? In RL, we can formalize this in terms of *behavioral rollouts* or *simulations.* Returning to the grid problems in Figure 5.1a, suppose an agent has a model of the world dynamics, $\hat{T}$, such that, for a given s, a, the agent can roughly predict which state it will occupy next, s'. Moreover we might assume this model is a sufficiently good model. That is, for every state of the world, the agent's prediction of what happens an action is executed is very close to the true model:

$$\max_{s,a,s'} |T(s' \mid s,a) - \hat{T}(s' \mid s,a)| \leq \varepsilon. \tag{4.5}$$

But, this is already problematic. Our agent must now retain a model of at least $|S|^2|A|$ parameter in its head for the above property to be true. No resource bounded agent could model anything remotely like this function. Things get even more complicated if the model is actually stochastic, as in games involving randomness like poker. This is simply further support for the use of good concepts: if an agent can only store K things in memory, then the function $\hat{T}$ better take less than K parameters to store. To do so, the agent should form more succinct characterizations of state and action that let the agent instead reason using $\hat{T}_\phi$, an abstract predictive model of the world. We will develop an account that supports such reasoning shortly.

But suppose our agent has unlimited memory (or that K is set so high so as to impose no real restriction). How might the agent use $\hat{T}$ to come up with a plan? Well, as suggested previously, the agent can use *rollouts*. Using rollouts for planning is effectively how Bill determined which airline to use. The agent closes its eyes in state s and simulates what would happen if it were to execute action a_1. Then, the agent simulates acting randomly for some number of time steps, leading to the eventual state s_h. The agent then evaluates the total utility of the path taken: $s_1, \ldots, s_h^{(1)}$. Then, it runs the same mental experiment again, this time executing a_2, leading to $s_1, \ldots, s_h^{(2)}$. The agent can repeat this process until it exhausts its computational budget – the action that received the highest *simulated utility.* This is the main idea behind tree-based planning methods, like the now popularized Monte Carlo Tree Search (MCTS) [34]. MCTS is a friendly formalism for planning in our circumstances since we can directly control the amount of computation required. An agent can



(a) Part of the standard search tree

(b) The search tree needed for the given state-action concepts

Figure 4.2: A comparison of the search trees used with different concepts for the grid problem in Figures 4.1c and 4.1d. In both cases, the agent plans forward from its start state shown in tan. On the left, the agent reasons about how its actions change the state of the world; it can first move to either the state above or state to the left. The full search tree is not pictured for brevity. On the right, we see the search tree formed if the agent has access to the action concepts that move it directly to the doorway ("move to the door"), which is clearly considerably more compact.

run R independent rollouts out to D depth in $R \cdot D$ computations. We can also get a sense of how many rollouts are needed so that $\hat{U}(s,a)$, the agents approximation of the utility of s in a based on the rollouts, well approximates $U(s,a)$, the actual utility.

Naively though, for this to work, agents would need far too many rollouts. Most agents have such a large state-action space that no reasonable budget would suffice. But, certain kinds of representations can make this dramatically faster; this is the main idea behind our present argument. **With a more succinct state-action space, with the same computational budget, an agent can run more rollouts deeper into the future, thereby getting a more accurate estimate $\hat{U}(s,a)$.**

For visual support for this idea, consider the search trees pictured in Figure 4.2. With some concepts, planning becomes more tractable. For further background and support in this direction, see the work of Zilberstein [176], Bernstein et al. [14], Zilberstein [177] and Harb et al. [59].

This is the sense in which state and action concepts directly impact an agent's ability to plan efficiently, subject to resource constraints. To conclude: if an agent A, does *not* make use of such state-action concepts with resource bound K, then there exists another agent, A^{+} that *does* make use of state-action concepts with resource bound K that always makes better decisions.

Moreover, we can combine the benefits of concepts that generalize and concepts for planning. Ideally, this would give us a representation of world state and action such that our model of the world (T) is compact, so planning is easy, and flexible, so that agents can adapt to new situations robustly. We already saw a notion of model compactness: the total number of s, a, s' triples the agent needs to remember can be smaller as the state representation becomes more parsimonious. What might it mean for this function to generalize? Intuitively, it means that the agent is equipped to make predictions about unseen world states. Matt, for instance, needs to reason about the consequences of mixing different baked ingredients together. This is still possible despite Matt never having actually used the exact kitchen of interest, to the precise destinations of interest, on this exact day at some point in his past. Instead, he is generalizing knowledge by making predictions based on similar experiences. So, imagine our agent has never before seen some state $\tilde{s}$. It must consider the prediction $T(\tilde{s}' \mid \tilde{s})$, for each action a and each possible next state of the world $\tilde{s}'$. How could an agent learn such a function? We saw in PAC learning how algorithms can learn, with high confidence, good approximations of functions based on finite data. We can exploit the same idea here: the agent sees some number of similar experiences (so, choices of airline), and uses that information to generalize to this unseen state. The remarkable thing is that *again*, the difficulty of PAC learning this function $T(s' \mid s, a)$ is going to depend on the size of the hypothesis space. In the simple case of a deterministic forward predictive model, there are $|S|^2|A|$ possible functions. With a smaller state-action space, learning this function again becomes easier. So: not only can the agent plan more with new concepts, it can learn the appropriate predictive model faster, too. In a similar vein, Harb et al. [58, 59] suggest that agents should adopt action concepts that make planning and learning simpler. Harb et al. define a new objective for finding action concepts subject to these considerations (often called "options", based on the formalism by Sutton et al. [156]), which leads to an algorithm for finding good action concepts.

The argument so far established suggests that there is a set of state-action concepts at the frontier of boundedly rational behavior–those representations that ensure no such other agent A^+ exists that dominates its behavior. In the next chapter, I argue that these properties define a notion of "good concepts" (and in turn give us a means of comparing concept sets) in the context of practical rationality.

4.2.3 Limitations

First, I consider a few natural counter arguments to those just presented.

1: Fast but useless planning In the previous section we saw how a smaller state space can lead to more planning in the same computational budget. But, there is no free lunch: surely we must lose something by diminishing our state representation! Most significantly, there is no guarantee that the plans we consider are actually reasonable in the sense of promoting high utility decision making. Indeed, the criteria we articulated suggests that we should just collapse the entire world into a single state! Planning is then maximally efficient.

Response: This is a crucial point. Not all smaller sets of concepts are created equal; we need those that are small (to make planning fast), generalize well (to work in a variety of domains) and useful (to preserve prediction of utility). Simultaneously satisfying these three criteria is a challenge, and I have here primarily motivated the first two properties. However, this third need not be forgotten. Indeed, in the next chapter, I will sketch a theory that tries to simultaneously satisfy all three properties. It is worth nothing, though, that there do exist methods of state (and action) representation that can still preserve utility [123, 93, 2].

2: What about other concepts? I began this argument by suggesting a decomposition of concept space into *state* and *action* representations, but surely this is an inadequate characterization of the conceptual landscape people actual rely on for decision making. What about the contents of introspection [148]? The contents of our introspection (or the vehicle by which we introspect) is likely not describable in terms of language about states or actions. Further, we have here been making the assumption that there is a true underlying world state, s, and that the agent observes it–in reality, agents just receive observations *based* on world state. It is not so clear that our concepts are describing world "state", so much as appearance properties of our observations.

Response: These are valid concerns; I here rely on the two simplifying (but wrong) assumptions that 1) concepts are fully exhausted by state-concepts and action-concepts, and that 2) agents observe s, not an observable that results from s. I take initial insights developed under these restrictions to be useful for a more fleshed out theory that relaxes these two key assumptions.

Chapter 5

A Compression-Based Theory of Rational Concept Formation

We have so far established two claims. First, that RL can give formal character to the problems faced by boundedly rational agents, and second, that talk of a resource-constrained agent's rationality must rely on the agent's concepts. We have yet to see, however, what it might look like for an agent to be *effective* in their choice of concepts. This chapter attends to this question. Specifically, like Carballo [21], I ask: "what constitutes a good choice of concepts"? Carballo's main goal is to address this question from the perspective of *epistemic* rationality. My goal is to answer this question from the perspective of *practical* rationality, cached out in terms of utility maximization in reinforcement learning.

To this end I will develop a theory of what makes a concept *better* than another. In principle agents are said to be rational (with respect to their choice of concepts) just when they develop mechanisms for adopting concepts that are better than all others. It is an open (and important) empirical question as to whether or not such an exercise is feasible for most people; surely our concepts develop over time as we attend school, learn about the world, and grow. But, it is unlikely that we have control of the nature of the concepts we use through introspection alone. It is hard to imagine that on reflection I can deliberate and actively decide to reason in terms of a particular new choice of concepts. Perhaps with the right training and strategy such a practice is possible, but again I leave this as an open empirical matter.

All that I set out to establish here is this: if an agent *did* have the ability to adapt its concepts, what might it look like for an agent to be justified in choosing new concepts?

We can answer this question by drawing on the technical machinery of RL. Like decision theory, agents that are effective at RL do a better job of maximizing utility. This same objective will lead us to an account of how agents can come up with concepts: those that support high quality sequential decision making.[1]

In general, what can we expect from our choice of concepts? Our answer emerged last chapter: fast, high quality planning, and robust generalization across different states of affairs and environments. This is not an exhaustive list. We regularly make use of concepts in order to learn lessons from our teachers and peers, to communicate our experiences, ask questions, and organize our daily lives [22]. Indeed, under resource constraints, our concepts are likely to be responsible for supporting many of these practices efficiently. I defer discussion of these properties and instead only focus on planning and generalization, as they are of sufficient interest for the argument to be useful. Since we have here restricted the scope of "concept" to state-concepts and action-concepts, we can rephrase our more general line of inquiry about concepts into something more focused. We ask: how can an agent choose its state and action concepts so as to support rational decision making, subject to resource constraints?

[1] This question is often studied in nearby learning settings such as representation learning [13] as studied in unsupervised [26], supervised learning [163], and even in "deep" RL [108].

By “choose”, we will mean that the approach for reasoning with state & action concepts can adequately order the utility of concept sets C_1 and C_2, for a given agent and the problems the agent faces. For example, consider two agents, $\mathscr{A}_1$ and $\mathscr{A}_2$. They both live in the same world, and have roughly the same goals and history. How might we compare the concept sets of these two agents? In our baking example from the previous chapter, we said an agent would need a variety of different concepts in order to bake a loaf of bread. Surely many are needed, like the concept of dough, and “to knead”. But what might it look like for a set of concepts to go beyond the bare necessities?

Before diving in, it makes sense to fix other properties of our agents, such as the stream of evidence they might go on to collect. In machine learning this is often referred to as *certainty equivalence* [69], and really just amounts to making sure we only vary the quantity of interest. For our present purposes, certainty equivalence will imply that agents given some data will act well according to the model they form based on that data. If we are baking, this just means that the agents choice of how to think about the kitchen will be informed by the same set of experiences (so: we might suppose our agents have each baked the same five cakes previously).

Recall that in RL, environments are usually assumed to be Markov Decision Processes (MDPs). For clarity, I here briefly review MDPs and related terminology; for more details, see Chapter 2 or Puterman [122].

Definition 7 (Markov Decision Process (MDP)): *A* **Markov Decision Process** *(MDP) is a five tuple:*

- $\mathcal{S}$*: a set of states describing the possible configurations of the world.*
- $\mathcal{A}$*: a set of actions, describing the possible choices available to an agent.*
- $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$*: a reward function, describing the task to the agent.*
- $T : \mathcal{S} \times \mathcal{A} \to \Pr(\mathcal{S}))$*: the transition function, which denotes the probability of arriving in the next state of the world after an action is executed in the current state.*
- $\gamma \in [0, 1)$*: a discount factor, indicating an agent's relative preference between near-term and long-term rewards. As* γ *gets closer to 0, agents prefer near term reward more.*

The "Markov" in **M**DP indicates that the transition function, T, and reward function, R, both only depend on the current state of the world (and action), and *not* the full state history. So:

$$\Pr(s_{t+1} \mid s_t, a_t) = \Pr(s_{t+1} \mid s_1, a_1, \ldots, s_t, a_t). \tag{5.1}$$

$$R(s_t, a_t) = R(s_1, a_1, \ldots, s_t, a_t). \tag{5.2}$$

In short: we can predict the next state distribution and next reward from *just* the current state and action. This is incredibly helpful for simplifying analysis. Moreover, if any problem is *not* Markov, we can always just roll the last k steps of the world into the state representation, thereby yielding a Markov model.

So, when we talk about certainty equivalence, we mean that different agents estimates of T and R will be the same, given the same data set. In this way, we make the default assumption that each agent will be rational according to whatever evidence it has gathered so far, given the resource constraints imposed, and given its concepts. This lets us ensure that the only free variables in the decision making processes involved are (1) the strategy the agents use to explore its environment, and (2) the state-action representations at play. This chapter is focused on (2), but let us quickly say a few words about (1).

Exploration is an active area of research, dating back to Thompson [160] and Bush and Mosteller [20], who introduced the first analysis of what is now known as the k-armed *bandit* problem or *hypothesis selection* problem. Exploration is a fundamental challenge facing any agent whose decisions effect their environment: learning about one's world (both the causal rules and the whereabouts of utility) while trying to maximize utility in that world is challenging. The key point of exploration is that agents do not get to observe the counterfactual of what *would* happened had they chosen a different course of action. So, it is hard to know whether the chosen behavior is a good one. I here refer the reader to recent approaches in exploration, such as Thompson sampling [24], Bayesian methods [120, 8], and approaches based on being optimistic in the face of uncertainty [68, 19]. Largely, though, exploration represents a major open question.

I speculate, too, that appropriate concept choice is needed for a truly general and effective exploration strategy. How does an agent know which experiments to conduct to learn about its environment? Per the "scientist in the crib" theory advanced by Gopnik et al. [55], infants conduct experiments in their world to falsify (or find support for) different theories. But which experiments should infants (and agents more generally) conduct in order to learn the most about their world? This is, in essence, just a rephrasing of the exploration problem. To choose among experiments, agents again ought to be equipped with appropriate concepts–how am I to know about what *could* be inside of a cupboard, without first having an appropriate concept of "cupboard" and the "inside of" relation or "hollow" property? Choosing which questions to ask about your world in order to learn the right things about your world is difficult, and something children seem to do remarkably well. Exploration is still entirely unsolved, however, and I don't claim to be contributing progress to its solution. I refer the reader to recent work in RL that suggests exploration and state representation may be connected [158].

For these reasons, we henceforth remain agnostic to approaches exploration as well. Therefore, we restate our dilemma, supposing exploration is *also* controlled for across agents: all agents explore in the same manner, and make the same (optimal) use of the given statistics. Then, we ask, what is the best choice of state and action concepts? "Best" here means "with the same resource budget, comes up with better decisions, as measured by expected utility."

The existence of such a general set of best concepts should not be taken as *a priori*. Without reaching into the world and collecting some evidence, no state-action concept choice can dominate all others. In other words, *a good state-action concept choice must depend on the actual problems and environments to be encountered by the agent that will use the concepts.*

Suppose for a moment this is not the case. Then there is one concept choice, C^*, that is better than all others for *every* environment subject to the resource constraints on the size of the concepts: $|C^*| \leq k$. But surely this can't be the case. Suppose we inspect an environment E. Any information contained in the concept set C^* that is responsible for making good decisions in *any* environment other than E could be directed toward improving learning on E. That is, we can construct a new concept set, C_1^*, that contains all of the useful information from C^* for learning in E, and remove anything else. Then, with the remaining space, we can add additional conceptual support that enhances the agent's learning on E. In this way, we can launch a No Free Lunch argument about the quality of the chosen concepts [175]. The result is that concepts *must* be defined in a domain-dependent way if they are to be good ones.

Our objective is then clear: to introduce a theory that can distinguish good state and action concepts from bad ones, where "good" means the concepts maximize the potential of an agent to make good decisions.

5.1 State and Action Concepts and Decision Making

Let us start with some intuition. Suppose you find yourself in a hallway. You look around, and notice a door at the end. Based solely on this information, which aspects of the hallway will take purchase on aspects of your representation of the situation? Psychological theories differ on which characteristics our cognitive representations pick up on, and further which properties we become aware of. Moreover, many aspects play into this processing: for instance, Li and Durgin [95] provide concrete evidence that people tend to poorly estimate certain quantities in our immediate environment (in this case, people tend to underestimate the slope of a hill in front of us). Even more peculiar, Bhalla and Proffitt [15] present evidence that people actually *overestimate* the incline of hills when they wear a heavy backpack. This view that "cognition impacts perception" is not without controversy, however—see Durgin et al. [39], Firestone and Scholl [43] for more detail.

However, surely *some* choices of concepts are better than others. Using the tools of learning theory, I will next show how to say something concrete about how different state-action concepts will effect learning difficulty, which brings us to our first point of the theory:

Good Concepts means Quick Learning: Agents should seek out concepts that make learning efficient.

If agents already *knew* everything about their environment, the above point would be moot. But, in the realistic case, no agent knows everything about its environment.

One might ask how we might measure learning efficiency. In the previous chapter, we introduced precisely the tool for this job: sample complexity.

5.1.1 Concepts Can Lower Sample Complexity

Recall that in the general sense of PAC learning, an agent seeks to choose a hypothesis $\hat{h}$ among a set of competing alternatives $\mathcal{H}$, that best explain the data. For our baking example, we'd like agents to decompose kitchens into the right fragments so as to make learning to use a new kitchen, or bake a new recipe, simple.

In the context of MDPs, the concept space we consider will directly define an agent's effectiveness. As a reminder, strategies for RL fall into one of three categories:

- **Model-Based**: The agent tries to learn R and T, which it can then use to plan by simulating based on its approximate model (as in R-Max).

  → Loosely, an agent tries to understand its world *first*, and then reason using this understanding to come up with behavior.

- **Value-Based**: The agent tries to learn V or Q directly, which are themselves sufficient for producing optimal behavior.

  → The agent learns to be reactive: when presented with two competing actions, the agent has developed an instinct that one tends to be better by some amount of utility (but may not know why they think that).

- **Policy-Based**: The agent tries to learn π^* directly, which itself encodes behavior.

→ The agent against follows instinct to take action (but doesn't necessary know the utility of the chosen action compared to others).

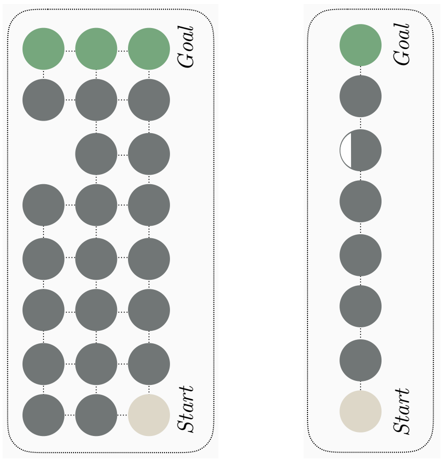
Note that value-based and policy-based methods are often grouped together under the name "model-free" methods. These three methods roughly cover the space of approaches to RL.

It is useful to consider this space, as we would like to inspect how concept choice will affect different strategies. Regardless of choice of these three strategies, we will reach the following conclusion: *an appropriately compact concept set makes learning faster, since any agent has fewer hypotheses to search through.* I next present a more technical version of this argument in the language of the PAC framework. However, note that this is a specific instance of the more general principle.

5.1.2 A Concrete Case

Consider again the hallway pictured in Figure 5.1. We see the original problem contains 23 states and 4 actions, while the variant with new state concepts has only 8 states, and 2 actions per state. So, for any of the functions the agent might want to learn (policies, transition models, and so on), there are simply fewer of them to consider.

Let us consider the class of **model-based** approaches to RL, in which the agent does its best to learn R and T, and use those to do forward planning. In other words, the agent sets out to learn a predictive model of its environment: given that the world looks like such-and-such, and I act according to this behavior, my goal is to predict how rewarding the next state of the world will be, and predict what the next state of the world will be. By the reasoning we have so far laid out, an agent will make these predictions in terms of its state and action concepts. What state comes next? It depends on how the agent represents states. What will the world look like after an agent



(a) Original Problem

(b) New State Concepts

Figure 5.1: Recall again the distillation of a hallway (left) into a long skinny problem (right).

executes one of its available actions? It depends on how the agent defines its action concepts. As I will next show, if the agent is learning with a more compact set of state-action concepts, R and T are actually easier to learn.

In the context of the hallway, a choice of state-action concepts restricts the hypothesis class over reward functions from $|\mathcal{H}_R| = |\mathcal{S}| \times |\mathcal{A}| * 2 = 184$, since each state-action pair can yield either 0 or 1 reward. For simplicity, let us instead restrict our attention to deterministic transition functions, thus inducing a hypothesis class of $|\mathcal{H}_T| = |\mathcal{S}|^2 \times |\mathcal{A}| = 2116$.[2]

However, using the appropriate state and action concepts, we can instantly lower the size of these two classes. In Figure 5.1, with the new state representation on the right, the hypothesis sizes now reduce to $|\mathcal{H}_R^\phi| = |\mathcal{S}_\phi| \times |\mathcal{A}| * 2 = 64$, and $|\mathcal{H}_T^\phi| = |\mathcal{S}_\phi|^2 \times |\mathcal{A}| = 256$. Clearly, the hypothesis spaces are now smaller.

Recall that we have assumed our agents all use the same exploration strategy. If we make the

[2] If we suppose the transition model can be an arbitrary probability mass function over next states, then there are infinitely many hypotheses, so we would require a more complex measure (like the VC-Dimension or Rademacher Complexity) to evaluate the complexity at play.

further (unrealistic!) assumption that the agent's experiences are all sampled independently, then by a straightforward application of Theorem 2.18, we know an agent can PAC learn T and R with:

$$\left\lceil \frac{\log \frac{|\mathcal{H}_R^{\phi}|}{\delta}}{\varepsilon} \right\rceil = \left\lceil \frac{\log \frac{64}{\delta}}{\varepsilon} \right\rceil, \tag{5.3}$$

$$\left\lceil \frac{\log \frac{|\mathcal{H}_T^{\phi}|}{\delta}}{\varepsilon} \right\rceil = \left\lceil \frac{\log \frac{256}{\delta}}{\varepsilon} \right\rceil, \tag{5.4}$$

samples, for our chosen accuracy and confidence parameters ε, δ.

For example, set ε and δ both to 0.05. Then, by using the smaller state and action concepts, we find agents with a difference in sample complexity of:

$$\text{Sample Complexity}(\phi, R) \leq \left\lceil \frac{\log \frac{64}{.05}}{.05} \right\rceil \approx 2400, \quad \text{Sample Complexity}(R) \leq \left\lceil \frac{\log \frac{184}{.05}}{.05} \right\rceil \approx 3010,$$
$$\text{Sample Complexity}(\phi, T) \leq \left\lceil \frac{\log \frac{256}{.05}}{.05} \right\rceil \approx 3200, \quad \text{Sample Complexity}(T) \leq \left\lceil \frac{\log \frac{2116}{.05}}{.05} \right\rceil \approx 4420.$$

In effect, we can shave off about 700 samples to PAC learn the reward function and over 1000 samples to PAC learn the transition function for any MDP (under the assumptions made above).

Why does shaving off samples matter here? Well, this translates directly to the *amount of evidence* an agent needs in order to retain a confident estimate of facts about the world. "Confidence" here is measured according to the PAC parameters, ε and δ, denoting the tolerable error (ε) and confidence (δ). The takeaway then is that an agent with better state and action concepts can actually acquire PAC-like knowledge of quantities that are *sufficient* for making high quality decisions.

To complete the argument, note that a PAC learned reward and transition function are in fact sufficient for supporting high quality utility maximization [19, 68]. In some sense this is the best a resource bounded agent can hope to do, since any practical agent will inevitably live in a finite data regime and so cannot hope to learn $\hat{T}$ that is equivalent to the true world dynamics T.

In fact, the same *exact* argument goes through for the other forms of learning. If an agent is instead trying to learn a value/utility function directly, the same reasoning holds: learning in a more abstract way means there are less hypotheses to search through, and so, it is easier to falsify a larger fraction of them with the same body of evidence.

Of course, in the real world, our experiences aren't so well behaved as to show up independently. They are connected: when we choose to take a job in Seattle instead of New York, we will never know what it might have looked like to accept the New York position. The above results violate this assumption. It turns out this isn't so problematic on the technical side: we can turn to more complicated measures, like Regret [98, 68] or the Sample Complexity of RL [79]. The conclusion, however, is effectively the same: learning to make good decisions is easier if we choose the right concepts for representing the state space.[3]

To conclude, we have seen how state-action concepts can profoundly affect learning difficulty. Moreover, by the same reasoning we established in Chapter 4.2.2, state-action concepts determine planning difficulty. Collectively, state-action concepts offer great potential for mitigating the difficulty of learning and planning, two fundamental practices of physically instantiated agents.

5.1.3 Accurate Concepts

We have so far glossed over one crucial point. As we make the set of hypotheses smaller, we might actually *throw away* all reasonable solutions.

This brings us to the second crucial property our state-action concepts must adhere to: they

[3] One might object that PAC bounds are loose upper bounds, and in practice agents would need considerably less data on average. Still, the tightest upper bounds tend to account for the worst case performance, so at the very least we are in a similar position to the concerns raised about computational complexity in Chapter 2.

need to support representation of good behavior. That is, they can't be so inaccurate such that an agent basing their decisions on them will be incapable of solving problems relevant to its own flourishing.

Returning to Matt and the baking example, suppose we give Matt state-action concepts that only make planning and learning easy. What might that look like? Unfortunately, the state-action concepts that reduce all world states and all behaviors to the same thing makes planning and learning trivial. There is nothing to learn and nothing to gain by planning. So, we have satisfied the first property.

Of course this is undesirable. Thus, we need to also require that the resulting set of hypotheses to be considered are actually useful. In the baking case, this says that not only can we translate what we knew before to a new kitchen, but that this translation *actually works*. In the more technical language, we might require that our new state concepts ensure that $R \in \mathcal{H}_\phi$. That is, the true function we are trying to learn is still representable in our new conceptual space. In Matt's case, the concept he uses have to carve out the world in a way that roughly latches on to some causal laws of the world (at least those that are directly involved in the kinds of predictions Matt will be making). This is not too different from Carballo's proposal about good epistemic concepts: they "carve the world at its joints".

But, ensuring that the true functions we care about are representable in terms of our concepts is not always trivial – we will often accidentally throw away the best hypotheses in our attempt to limit the hypothesis size. This is precisely the difficulty of choosing the right state and action concepts: the smaller our resulting hypothesis space, the easier it is to learn to make good decisions (across each of the three approaches to learning), but if we make it too small, we might throw away too many good candidates. But if the hypothesis space is too large, then finding a good hypothesis is challenging. If the hypothesis space were a hay stack filled with some needles, the agent is searching through it to find the needles. If there is too much hay, the agent has no hope. Our trimming of the hypothesis space tries to control the needle-to-hay ratio so as to make the search efficient, while not throwing away all the needles.[4] We might try to impose constraints on the hypothesis space to get rid of some hay (but in so doing we may get rid of a bunch of needles!).

This is in effect the fundamental problem of machine learning. The recent learning textbook by Ben-David and Schwartz says of the matter: "Learning theory studies how rich we can make $\mathcal{H}$ while still maintaining reasonable estimation error" [132]. In RL, matters are made even more difficult as agents must explore and solve credit assignment, in addition to the usual problem of generalization.

We have thus uncovered a dilemma facing agents: how can they form the appropriate state and action concepts that ensures both of the following properties?

1. *Concepts Make Learning and Planning Easy:* learning is easier (because the hypothesis space is compressed) and planning is easier (because our plan space is compressed).

2. *Concepts Support Accurate Predictions:* the hypothesis class contains enough relatively accurate entities so that the agent can learn to make good decisions.

The thesis of this chapter is that good state-action concepts are responsible for trading-off between maximizing both of the above properties. In essence, this trade-off raises a *problem*, which I henceforth call the Practical Concept Choice (PCC) problem, that asks how resource bounded agents should be making the above trade-off. This problem is intimately related to (and indeed, heavily inspired by) Zilberstein's proposal of metareasoning through bounded rationality [178]; on

[4] We could throw away all the needles and still make the search efficient by making it trivial to determine there are no needles!

his view, one strategy for approaching bounded rationality is to call on optimal metareasoning that figures out how best to spend available resources. This in turn sets up tension between utility and computation. The PCC problem is intended to build on this view by extending it to concepts: we have seen how, under restricted assumptions, certain concepts can lead to more efficient learning, but at the same time can throw away good solutions. A bounded rational agent makes a choice as to how it must trade-off between these two factors.

I suggest that this question stands as an open and important piece of our broader story about rationality.

In the next section I propose an initial story for how this trade-off might be made (and thus draft an initial solution to the PCC problem). To do so I lean on information theory, and more specifically, to rate-distortion theory, which presents a mathematical framework for studying trade-offs of the relevant kind.

Many of the insights explored in this chapter are rooted in the earlier work of Ortega Jr [117], Ortega and Braun [116], who first proposed an information theoretic framework for studying bounded rationality. Indeed, Ortega's approach *also* calls on decision theory (and to some extent, RL) to formalize the problems facing a bounded rational agent using tools of information theory.[5] The core of Ortega et al.'s method forms a new objective that forces agents to explicitly trade-off between solution quality (expected value of the chosen decisions) and the *information cost* associated with the decision. Intuitively, the information cost measures how complex the choice of decisions is, relative to some initial prior choice of decisions. Ortega models this deviation according to the Kullback-Leibler divergence between an agent's chosen prior and the new behavior the agent is likely to adopt [87]; in this sense, a similar trade-off is being optimized for. I take Ortega et al.'s

[5] Rubin et al. [127], Tishby and Polani [161] has also explored connections between information theory and RL.

view and the account developed here to be entirely complementary to each other. The focus of this investigation is on trade-offs of this kind in the space of *concepts* an agent chooses, as opposed to the metareasoning involving minimizing something like the information cost of behavior. For more in this direction, see the work of Harb et al. [59], which proposes a particular method for choosing action concepts that brings together deliberation costs with concept choice.

5.2 Concepts

What we are after are concepts that satisfy two properties:

1. SmallConcepts: The first property says that the smaller our set of concepts is, the more efficient learning and planning can be.

   → For learning: we saw this above with the hallway example. The upper bound on worst case samples needed to (PAC, or otherwise) learn the relevant concepts goes down *directly* as the concepts become simpler.

   → For planning: we saw this in the previous chapter. With more succinct concepts, the same planning budget lets the agent get away with planning farther into the future, and some cases, more widely, too (since we collapse separate plans into a single plan).

2. ActionableConcepts: The second property says that we still want to be able to find concepts that track enough with reality so as to support good decision making. In other words, if the agent is learning behavior directly, we would like the best learnable behavior to still achieve high utility.

   → This can be measured mathematically in terms of properties like the *value loss*, which measures how much value a given behavioral policy, $\pi_{\phi,\mathcal{O}}$ can achieve in the given MDP,

M. Recall that $V^\pi(s)$ denotes the *expected discounted value* that policy π will achieve when executed starting in state s. Using this definition, we can define the loss of a policy expressed in terms of the new state-action concepts, $\phi, \mathcal{O}$, as follows:

$$L_M(\pi_{\phi,\mathcal{O}}) \coloneqq \max_{s\in\mathcal{S}} |V_M^*(s) - V_M^{\pi^*_{\phi,\mathcal{O}}}(s)|. \tag{5.5}$$

If the above quantity comes out to zero, then the concepts represented by ϕ (state) and $\mathcal{O}$ (action) can *perfectly* represent optimal behavior. As the loss increases, it means our state-action concepts throw away more important information for making good decisions.

To get a true sense of how much decision-making quality one might lose by using new concepts, we need to measure the value loss of the *best* behavioral policy representable in terms of the new concepts:

$$\min_{\pi_{\phi,\mathcal{O}}} L_M(\pi_{\phi,\mathcal{O}}) \tag{5.6}$$

As discussed in the previous section, these two properties constantly pull on one another, giving rise to the PCC problem. If we make the concept space small, then planning and learning are easy. But, if we make it too small, we fail to preserve representation of good behavior. Really, rational agents need both. How can agents properly coordinate between these two forces?

Intuitively, I will propose a strategy for seeking out concepts that are as *compressed* as possible while still facilitating representation of good behavior. We can choose between concepts C_1 and C_2 by inspecting which better trades-off between SmallConcepts and ActionableConcepts.

Of course, different situations might call for different priorities: in a case where decision making needs to be extremely precise, then ActionableConcepts is likely given more weight. If C_1 and C_2 are equal according to their measurement in SmallConcepts, then C_1 is said to be *better* just when C_1 dominates C_2 according to ActionableConcepts (and vice versa). To find a mechanism for making this evaluation, we turn to information theory, and in particular, to recent work in choosing good state concepts based on their capacity to adhere to SmallConcepts and ActionableConcepts [3].

Before getting lost in the weeds, the main point of the next section is roughly as follows: there is a principled method for finding *state* concepts that adhere to SmallConcepts and ActionableConcepts as best as possible. The major caveat is that we require as input a parameter, β, that will determine an agent's preference between these two properties. If β is 0, then the agent only cares about SmallConcepts, while as β grows closer to ∞ the agent only cares about ActionableConcepts. For a given choice of the parameter, the method we describe will make the appropriate trade-off. No general procedure for determining how to best make this trade-off is yet known, but represents an open area for future work.

The theory I articulate is about state concepts. Harb et al. [59] presents a similar idea for action concepts, but is based on deliberation cost rather than compression. Understanding the nature of good action concepts is an active area of research in RL [66, 105, 106, 150, 141, 142, 143, 84, 99, 83, 100, 101, 42, 113, 125]. I take it as a given that some nearby scheme for finding action concepts (together with state concepts) will go through in the relevant sense.[6]

5.3 Information Theory to the Rescue

Information Theory studies communication in the presence of noise [133]. In Shannon's words: "The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point." Information Theory typically investigates

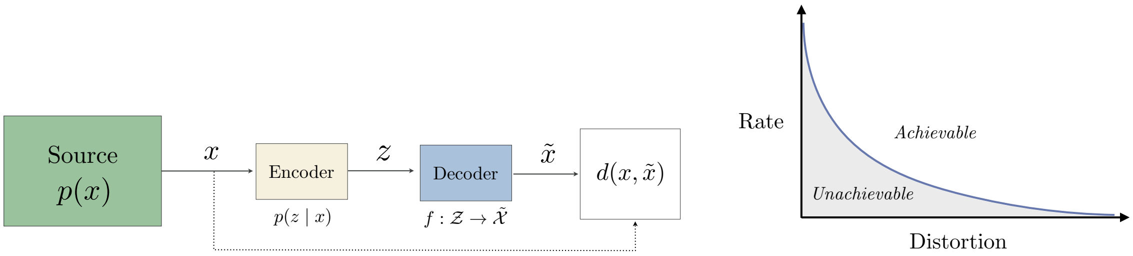
[6] Some approaches also exist for jointly finding state-action concepts, instead of finding them separately [123, 74, 31, 10, 23, 159, 103].

coder-decoder pairs and their capacity to faithfully communicate messages with zero or low error, even in the presence of noise. For more on information theory, see [133, 35].

Why call on information theory? Well, what we are after is a way to explicitly model the trade-off made by a set of concepts between ActionableConcepts and SmallConcepts. Information theory tells us how small things (broadly construed) can get before throwing away too much information. As discussed previously, Ortega and Braun [116] presents a bottom up account of information theory in bounded rationality.

In particular, we will turn to rate-distortion theory, which studies the trade-off between the compression and faithful reproduction of the original signal. The typical RD setting is pictured in Figure 5.2a: an information source generates some object, say an image of a landscape, or a piece of music. We then *encode* this object by writing down some short description of the object called the “code”. This code is supposed to be as *small* as possible—this is where the compression takes place. Then, using this code, we need to *reconstruct* the original object (image of the landscape, the song, and so on). In this way, what we are looking for is the coding-decoding scheme that can lead to small codes while still enabling those codes to produce accurate images. “Rate” is defined as the number of bits in the code, and “distortion” is defined by how much the input object and the reconstructed object differ. If we only ever had to reconstruct one object (a single landscape image) then our code could be arbitrarily small: the solution to always reproduce the original input image will work just fine. But, if we have many potential objects that we need to be responsible for coding and reconstructing, then our code needs to be quite a bit longer to capture the relevant distinctions between objects.

In this way, if there are structural similarities across objects, they can be captured by the same elements of the code. For example, if we have to code many different landscape images, one bit



(a) RD Setting (b) RD Lower Bound

Figure 5.2: The usual Rate-Distortion setting (left) and the lower bound on the Rate-Distortion trade-off (right)

of our code might be the color of the sky: if it is blue, then the "sky-color" bit is a 0, if it is a dark, night-time sky, then this bit is a 1. Thus, *codes* are really responsible for breaking apart the structure inherent in the objects of interest.

It is here the concepts start to emerge. Which concepts should be used to break apart the structure inherent in the entities we must code? Shortly, we will move away from the process of *reconstruction* (which is more in line with epistemic rationality), and onto the process of *decision making*, which connects directly with our overall goal of practical rationality.

5.3.1 Finding the Right Codes: Blahut-Arimoto

The work ahead of us will be to leverage the insights from rate-distortion theory to give an account of good concepts. To do so, what we really need is a sense of how to *find* the coders/decoders that make the right trade-off.

Ideally, we would come up with a method that can determine the appropriate preference between these two quantities based on the problem. Perhaps in the given situation the agent needs to make decisions extremely quickly, and would be better served with more compact concepts.

In 1972, Blahut and Arimoto independently discovered an algorithm for finding these optimal coder-decoder pairs [16, 6]. Their algorithm (now called the Blahut-Arimoto algorithm) finds the coder-decoder pair that perfectly solves the trade-off between rate and distortion. To be more precise, the algorithm takes as input a number, $\beta \in [0, \infty)$, that indicates a relative preference between compression (rate) and accuracy (distortion). If $\beta = 0$, then the algorithm only prioritizes compression, and so distortion will be arbitrarily high. As β grows, the algorithm finds coder/decoder pairs that favor distortion.

To summarize, for a particular choice of *how* to make the trade-off, for a particular set of objects (landscapes, songs, and so on), Blahut-Arimoto will find a coder-decoder that *optimally* makes the relevant trade-off. That is, there is no coder/decoder pair that does a better job of making the trade-off, for that particular β.[7]

However, critical to RD theory is the *reconstruction* of the input signal. Take another look at Figure 5.2a. At the end of the pipeline, we do our best to *rebuild* the original signal x. If all we wanted our agents to do in the world was reconstruct some input signal, then BA would give us our solution.

Let us turn to thinking about practical rationality once more: to what extent does an agent's capacity for making good decisions depend on reconstruction of its stream of perceptual data? It is likely that the appropriate answer is: not at all. Of course, what we *do* with our perceptual data is not on its own uncontroversial. But, for our purposes, it is unclear how to relate reconstruction of percepts with good decision making.[8] However, in the realistic setting in which agents are cognitively limited by resource constraints, it is hard to justify allocating lots of cognitive resources for

[7] It is worth saying a few words about the speed of the algorithm: BA is known to converge to the global optimum with convergence rate:

$$O\left(|\mathcal{X}||\tilde{X}|\sqrt{\log(|\tilde{X}|)}/\varepsilon\right), \tag{5.7}$$

for ε error tolerance [6]. The computational complexity of finding the exact solution for a discrete, memoryless channel is unknown. For a continuous memoryless channel, the problem is an infinite-dimensional convex optimization which is known to be NP-Hard [153].

[8] Perhaps reconstruction is sufficient for making good decisions, when paired with other hefty machinery that does the appropriate thing with those perceptual data.

the purpose of reconstructing precise perceptual input. Those same resources could be better spent on remembering critical things from the recent past, modeling laws relevant to making accurate predictions of the environment, or on supporting better state/action concepts. So, I take it that we are not really after reconstruction, but instead, on making good *decisions* based on a compressed concept space.

Our next and final move is to convert the approach taken by Blahut-Arimoto to one that incorporates decision making as opposed to reconstruction.

5.3.2 Reconstruction to Action: The Information Bottleneck Method

We will turn to a recent extension of RD theory that explicitly incorporates decision making: the Information Bottleneck method (IB). As discussed, traditional RD theory defines "relevant" information by choice of a distortion function—codes are said to capture relevant information if they achieve low distortion. The IB defines relevant information according to how well the given code can support *prediction* of relevant properties about the given object.

So, for example, suppose again we are coding images of landscapes. Instead of decoding our code into an image that is supposed to approximate the original landscape, we now ask: what questions can I answer using *only* the code? Can I determine that this particular landscape is of Italy? That there are mountains? That it depicts a sunrise? If we can ask questions of this form using the code, then our code is said to capture the relevant information about the original image. This way, we can avoid wasting precious cognitive resources on reconstruction of unnecessary details ("how many leaves are depicted?").

Tishby et al. [162] offer a convergent algorithm for solving this extension. Specifically, they prove that there is an extension of the Blahut-Arimoto algorithm to the case of prediction. The only difference is that the new algorithm is not guaranteed to find the *perfect* solution, but instead a *reasonable* solution. To the best of our knowledge, there is no known efficient algorithm for computing the global optimum. Mumey and Gedeon [111] show that a closely related problem to finding the global optimum in the above is in fact NP-Hard, suggesting that local convergence or approximation is likely our best option.

Let us recap the newly introduced technical tools: we set out to find a means of comparing concept sets based on their rating of ActionableConcepts and SmallConcepts. I have now proposed the information bottleneck method as formal means of studying the trade-off between the extent to which a set of concepts satisfy these two properties. A concept set is encouraged to be smaller by lowering its size, but still retain some accuracy by retaining its ability to make predictions about some relevance variable y.

We now extend these insights to RL by supposing that the relevance variable y is consists solely of *good decisions*.

5.3.3 Extension to RL

For the IB to be useful for our purposes, we need it to extend beyond simple prediction to the more general case of sequential decision making.

This gives rise to a particular model for finding state concepts, pictured in Figure 5.3. Recent work presents an efficient algorithm for producing state concepts in this setting [3].

To see the idea, let us return to Matt's baking endeavors in the kitchen. The premise of the algorithmic structure we have so far sketched is that Matt can *watch other agents* perform various baking related tasks (presumably those agents good at baking) and identify when other agents take actions. Those world states that are important for determining when other bakers do different things are crucial for ensuring that Matt, too, can bake like these other agents. The other source of information Matt needs is his own reward signal: did the bread taste good after I made it? What contributed most to its tastiness? Any concepts needed to answer these questions will be crucial as well.

Philosophically, this means that agents should update their concepts so as to either 1) make them smaller (in accordance with SmallConcepts) or 2) make them support better decisions (in accordance with ActionableConcepts).

The key insight of the algorithmic work referenced above is that we need to ensure that an adjustment of either kind is worth it. We would hate to make our concepts simpler only to completely destroy our ability to make good decisions!

One relatively desirable feature of this theory is that the evidence agents receive directly suggests updates of either of the two kinds. Specifically, in learning, agents receive evidence of two distinct forms:

1. Lessons or demonstrations provided by teachers: either an agent is privy to a demonstration or a teacher provides ostensive instruction.

2. Reward: evolved agents receive internally generated reward to elicit repetition of behaviors that tend toward survival and reproduction [4, 131].

In the framework described, agents get to watch a teacher/expert (or any more experienced agent) make decisions. They then search for state concepts that can still explain what the teacher does. Crudely, the algorithm performs updates by carrying out one of two operations: collapse the state concepts to be smaller by grouping together similar states (those where the teacher behaves similarly), or break apart states in order to ensure that relevant distinctions in good behavior can

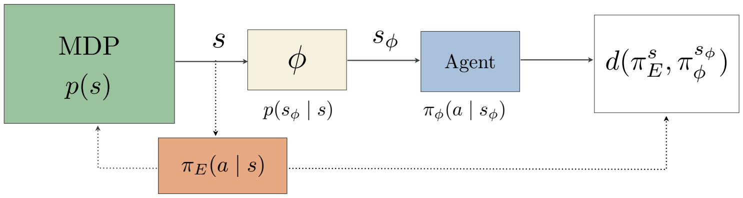


Figure 5.3: Trading off compression with value via well chosen state concepts. We suppose the MDP generates states according to some probability distribution, $p(s)$. Then, the agent forms its state representation, s_ϕ, and chooses an action from its policy π_ϕ. We then inspect the *distance* between what an expert's behavioral policy would do, $\pi_E(a \mid s)$, and what the agent would do. The goal here is for the agent to form state concepts that can still represent roughly what an expert does, in the states generated by $p(s)$.

be made. Threshold for state similarity and relevance are determined according to the parameter β, without which the objective is under specified.

For an individual, the idea is that we observe our companions within our community and our own reward signals to determine our concepts. There is now strong evidence that people (and indeed, many other mammals) have evolved strong neural mechanisms for predicting reward and punishment [131]. Using this reward prediction mechanism, agents can hone concepts that do a good job of supporting long-term expected reward prediction. In the context of the introduced algorithm we'd imagine an agent learning which concepts lead to distinctions in their own reward, or in prediction of other agent's behavior – by repeatedly performing updates of this kind to one's own concept set, one would eventually find a parsimonious, but still accurate set of concepts.

To summarize: using these two sources of evidence, boundedly rational agents can hone their concepts by searching for the simplest set of state-action concepts that still fit the data. The work discussed in this chapter presents a partial path toward realizing such a theory.

5.4 Objections

I now discuss objections facing the theory.

5.4.1 O1: Invalid Assumptions

First, one might take issue with the setup: the theory is overly specialized given the assumptions. More over, we set out to move away from the unrealistic assumptions of ideal rationality, and ended up in an even worse off position (regarding realism). Perhaps the worst of these are assumptions that state and action concepts can be updated (attended to in O2), the Markov assumption, and that the remaining aspects of the agents in question are held constant (planning, exploration, and so on).

Response

To get any theory off the ground we need to make some simplifying assumptions – I take the scope is still of sufficient breadth so as to be useful. Naturally, future work involves relaxing these assumptions. To summarize some of them, we required a Markovian world model (underlying the MDP), assume discrete state and action spaces, a fixed discounting rate γ (that determines an agent's preference for immediate vs future rewards), and only deal in state and action concepts. Of course, each of these adds convenient simplifications to our model that might be relaxed in future work. To say a few initial words on these relaxations: the Markov property can be surprisingly general if we allow for the world history to be folded into the state representation (which also then allows all results on MDPs to extend to these more general history-based processes). The discount factor is notoriously convenient-but-wrong in the RL literature. However, alternate formulations of the objectives RL exist, like the average reward criterion. And lastly, state and action concepts still span a rich and important subset of all concepts, so I take them to be of sufficient interest.

5.4.2 O2: Impracticality

To frame our next objection, let us return to the following question: what is the purpose of studying (and creating explanatory theories of) rationality? We had stated that one answer might be that it is profoundly useful to have a clear picture of good reasoning to act as a guide to better understand how we *should* behave, given a variety of circumstances. If we can form a simple explanatory view of rational behavior, it is likely that we can use this view to prescribe how to behave.

Now that concepts are in the mix, the same expectation should be placed on a theory of rational concept formation. Such a theory ought to be practical enough to act as a guide for clarifying our own concept acquisition and exploitation practices. However, one might here object: since we are not directly in control of the set of concepts we use to reason with, such a theory fundamentally cannot act as a guide since we are incapable of incorporating its advice.

Moreover, our concepts by default are good enough (most people can navigate a city, write a letter to a friend, or play a game of monopoly – all diverse and seemingly challenging feats). Why care about making slight improvements to them? What can we really gain by making changes to concepts?

Response

It is largely an empirical matter as to whether people can in principle update their concept sets in the relevant sense. This is largely a new proposal, so perhaps with the right tools and techniques, improvements can be made to one's concepts. As suggested previously, by better understanding the processes underlying efficient learning and decision making, we may uncover new avenues for individuals to learn efficiently in the real world. Still, even if individuals can't explicitly update their own concepts (quickly, at least), we might still find it useful for determining whether or not concepts are useful in a particular domain. Clarifying the essence of rationality is still of interest and use; in part, we went from a position of highly unrealistic assumptions (as in ideal rationality) to an account with potentially unrealistic assumptions. So, we have moved the view of rationality to one that is closer to the right picture.

To the second point, it is an open empirical question as to how much more useful our concepts can become. Simulations and the PAC-like insights presented in the previous section tell us that good concepts can dramatically improve an agent's ability to efficiently learn to make good decisions. Perhaps our concepts for most day-to-day tasks are sufficient, but by better understanding the process of good concept acquisition, individuals might be able to better specialize and learn new skills or domains by concentrating explicitly on learning good concepts. Even if there are some cases where we gain in utility from improving concepts, that is sufficient to motivate the theory.

5.4.3 O3: Concept Size

No measure of concept size from RL has a useful analogue in humans. So, why bother with addressing the notion of concept size?

Response

To see whether this argument has weight, let us summarize the methods for capturing concept size in RL. In the simple case of measuring concept size in terms of state concepts, we find the following reasonable candidates:

- The number of propositions needed to encode relevant state configurations.
- The number of bins involved in the abstract state space: $|\mathcal{S}_\phi|$.
- The number of bits required to write down the mapping ϕ.

Of course, as with any model, much of the formal framework depends on critical assumptions that don't hold in the real world. However, I stipulate that there is enough traction between the above quantities (and likewise for action concepts) that we can at least built out our initial theory using them, and clarify further as needed. None of the above measures are perfect, but they largely track with what we mean by concept size. It is partially an open question as to how best measure the size of any computational object, drawing on many of the same issues that emerged when we discussed shortcomings of worst-case complexity analysis in Chapter 2. Additionally, we can in part defer to the psychological literature for a clear and practical picture of concept size, and assess the degree to which these track based on our best estimates from current methods. Thus, I don't think this objection carries much weight. It is still extremely useful to understand how the size of concrete objects interplays with planning and learning difficulty, even if the units of measurement are a slight deviation from what we will actually use to measure aspects of cognitive practices of biological organisms.

5.4.4 O4: Fixing β is Too Restrictive

As stated, the core algorithm requires that we know up front the right trade-off between SmallConcepts and ActionableConcepts. Even worse, the *units* for these two quantities isn't even the same, so β is measureless in some sense (and must be drawn from the open interval $[0, \infty)$). The vast majority of the work is still to be done: what we really need is a theory for dynamically determining how much to prioritize each property based on the current situation.

Response

This is indeed a practical limitation of our current algorithm. However, the structure is suggestive of a broader idea: agents should be making this trade-off in the right way. It is an open technical question as to how to do this precisely when β is not known, but I don't take this to be a direct knock on the broader theory introduced. It just means there is more work to be done. Besides, existing work in metareasoning, such as the early work of Zilberstein [176, 177] and Horvitz [62, 64] tackle precisely the sorts of problems involving choosing β. It is useful to know how we might make this trade-off for a fixed β, and is solid first step in generating the more robust theory that can discover the appropriate β, possibly by relying on existing techniques from the metareasoning literature. So, again, I take the initial theory offered here to be useful, and to be suggestive of more general analysis for thinking about resource constrained rational decision making.

Chapter 6

Conclusion

This work first set out to give formal character to rational agents that must act under resource constraints, building on Simon's bounded rationality and the subsequent developments from Gigerenzer's ecological rationality, computational rationality [49], metareasoning [177, 62], and other relevant frameworks [117, 59]. I first argued that RL can serve as the right kind machinery for studying rationality of these kinds (that is, under realistic assumptions). In RL, agents interact with an environment they initially know nothing about. Through this interaction these agents collect evidence that offers insight into the nature of the world they inhabit and the goals they seek. The problem facing these agents is to learn about their world in this manner while using collected evidence to inform their decision making. As discussed in Chapter 3, to get off the ground, RL research tends to make several key simplifying assumptions. For our purposes, we find these assumptions are largely agreeable; the resulting formalism is still useful enough to offer new insights, even if the assumptions are ultimately unrealistic. Most significantly, RL gives a unique vantage to unify the process of gathering evidence (the "exploration" problem), perception (which aspects of the world state does the agent actually base its decisions on?), planning (does the agent reason about future consequences to inform its decision?), and utility-maximizing action (how much utility did the agent's achieve in its course of action?). All the while, agents studied under this unifying perspective can be given *explicit* resource budgets in the form of computational resources like time and memory. I conclude from these facts that RL is a useful method for investigating rational decision making under resource constraints. For more on this argument, see Chapter 3.

We next turn to the primary contribution of this work: understanding the role that *good concepts* play in bounded rationality. This inquiry constitutes Chapters 4 and 5. To summarize these arguments, let us consider two agents that are identical apart from their concepts. After making some simplifying assumptions, we showed that the agent with the correctly chosen set of concepts will be more effective at making decisions than the other, as assessed by the total amount of expected utility the agent can achieve. In so far as expected utility maximization is an appropriate goal for practical rationality, concepts can change an agent's potential for making good decisions, and in turn, are critical to any view of practical rationality.

Chapter 5 dives deeper into what constitutes a *good* concept. Under the broader goal of practical rationality, we want concepts that support high utility decision making. The problem we uncovered suggests that concepts can contribute to making better decisions if they are either *compressed* (so that planning and other internal computation is easy), or are *accurate* (so that predictions based on them track with the world). We formalize these two properties as SmallConcepts and PracticalConcepts respectively. Unfortunately, these two properties are in fundamental tension with one another. As an agent's concept set becomes more parsimonious, it loses expressiveness. So: bounded rational agents face a fundamental dilemma about how to best trade-off between these two conflicting properties. I claim that this dilemma is central to any study of rationality under realistic assumptions. I conclude by giving an initial response to the dilemma based on ideas from information theory and rate-distortion theory; agents receive evidence that lets them update their concepts so as to make them smaller or more accurate (possibly at the expense of one another). A boundedly rational agent, then, is one that makes this trade-off effectively. I sketched an initial pass as to how an agent might make this trade-off based on recent work in RL. I close by raising and responding to several objections to the theory.

Many open questions remain, both theoretical and empirical. Perhaps the most pressing technical question is whether a more general variant of the algorithm presented in Chapter 5 exists that does not explicitly require a predefined trade-off between parsimony and accuracy. I suspect ideas from metareasoning might prove useful here. Second, there is still a profound open question as to how to measure learning difficulty in general. PAC learning and the broader statistical learning framework laid the groundwork for measuring how hard learning certain functions is, under different assumptions. However, there is as of yet no consensus for measuring the difficulty of an arbitrary *problem* in the RL sense of the word. As discussed in the background, we have at our disposal many tools for measuring computational difficult of decision problems (and their kin). However, being effective at RL is not just about raw computation. It involves exploring, generalizing, transferring knowledge, handling uncertainty, and planning with an imperfect model – all of which together pose difference *kinds* of challenges. Recent work has proposed measures of hardness [102], but there is still room for a more general measure. The theory I introduced suggests that parsimony alone may guarantee that learning is easy. However, this is not always the case. Clearly there are large but easy problems and small but hard problems. To establish a theory for defining concepts that trade-off between the relevant properties, we need a true measure of learning difficulty.

In summary, RL has a lot to offer the study of rationality because of its close attachment to (nearly) realistic assumptions of agents that learn to solve problems. Our immediate and most significant finding was that the concepts an agent uses to reason about its objectives and plans actually plays directly into its ability to be rational. I take the initial foundations established here to suggest that effective concept formation might be thought of as a rationally required project [41]. One consequence might be that people should think more about explicit strategies for building good concepts, or at the very least, we can identify aspects of our concepts that are ineffective (such as those that are uncompressed). Or, alternatively, to better understand how one should learn about a new game, subject, hobby, or otherwise, perhaps it is important to appropriately emphasize learning the right concepts, first.

Bibliography

[1] Scott Aaronson. Why philosophers should care about computational complexity. *Computability: Turing, Gödel, Church, and Beyond*, pages 261–328, 2013.

[2] David Abel, D. Ellis Hershkowitz, and Michael L. Littman. Near optimal behavior via approximate state abstraction. In *Proceedings of the International Conference on Machine Learning*, pages 2915–2923, 2016.

[3] David Abel, Dilip Arumugam, Kavosh Asadi, Yuu Jinnai, Michael L. Littman, and Lawson L.S. Wong. State abstraction as compression in apprenticeship learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[4] David Ackley and Michael Littman. Interactions between learning and evolution. *Artificial life II*, 10:487–509, 1991.

[5] David Andre and Stuart J Russell. State abstraction for programmable reinforcement learning agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 119–125, 2002.

[6] Suguru Arimoto. An algorithm for computing the capacity of arbitrary discrete memoryless channels. *IEEE Transactions on Information Theory*, 18(1):14–20, 1972.

[7] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[8] John Asmuth, Lihong Li, Michael L Littman, Ali Nouri, and David Wingate. A bayesian sampling approach to exploration in reinforcement learning. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 19–26. AUAI Press, 2009.

[9] Robert J Aumann. Rationality and bounded rationality. In *Cooperation: Game-Theoretic Approaches*, pages 219–231. Springer, 1997.

[10] Aijun Bai, Siddharth Srivastava, and Stuart J Russell. Markovian state and action abstractions for MDPs via hierarchical MCTS. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 3029–3039, 2016.

[11] Hannah M Bayer, Brian Lau, and Paul W Glimcher. Statistics of midbrain dopamine neuron spike trains in the awake primate. *Journal of Neurophysiology*, 98(3):1428–1439, 2007.

[12] Richard Bellman. Dynamic programming and lagrange multipliers. *Proceedings of the National Academy of Sciences*, 42(10):767–769, 1956.

[13] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828, 2013.

[14] Daniel S Bernstein, Eric A Hansen, and Shlomo Zilberstein. Bounded policy iteration for decentralized pomdps. In *Proceedings of the Onternational Joint Conference on Artificial Intelligence*, pages 52–57, 2005.

[15] Mukul Bhalla and Dennis R Proffitt. Visual–motor recalibration in geographical slant perception. *Journal of experimental psychology: Human perception and performance*, 25(4):1076, 1999.

[16] Richard Blahut. Computation of channel capacity and rate-distortion functions. *IEEE transactions on Information Theory*, 18(4):460–473, 1972.

[17] Lawrence E Blume and David Easley. Rationality. *The new Palgrave dictionary of economics*, pages 1–13, 2016.

[18] Andrej Bogdanov, Luca Trevisan, et al. Average-case complexity. *Foundations and Trends® in Theoretical Computer Science*, 2(1):1–106, 2006.

[19] Ronen I Brafman and Moshe Tennenholtz. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3(Oct):213–231, 2002.

[20] Robert R Bush and Frederick Mosteller. A stochastic model with applications to learning. *The Annals of Mathematical Statistics*, pages 559–585, 1953.

[21] Alejandro Perez Carballo. Conceptual Evaluation: Epistemic. *Conceptual Ethics and Conceptual Engineering*, 2018.

[22] Susan Carey. *The origin of concepts*. Oxford University Press, 2009.

[23] Pablo Samuel Castro and Doina Precup. Automatic construction of temporally extended actions for MDPs using bisimulation metrics. In *EWRL*, 2011.

[24] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. In *Advances in Neural Information Processing Systems*, pages 2249–2257, 2011.

[25] Nick Chater. The search for simplicity: A fundamental cognitive principle? *The Quarterly Journal of Experimental Psychology: Section A*, 52(2):273–302, 1999.

[26] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2172–2180, 2016.

[27] Christopher Cherniak. Minimal Rationality. *Mind*, 90:161–183, 1981.

[28] David Christensen et al. *Putting logic in its place: Formal constraints on rational belief.* Oxford University Press on Demand, 2004.

[29] Brian Christian and Tom Griffiths. *Algorithms to live by: The computer science of human decisions.* Macmillan, 2016.

[30] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

[31] Kamil Ciosek and David Silver. Value iteration with options and state aggregation. *arXiv:1501.03959*, 2015.

[32] Stephen A Cook. The complexity of theorem-proving procedures. In *Proceedings of the ACM Symposium on Theory of Computing*, pages 151–158. ACM, 1971.

[33] James E Corter and Mark A Gluck. Explaining basic categories: Feature predictability and information. *Psychological Bulletin*, 111(2):291, 1992.

[34] Rémi Coulom. Efficient selectivity and backup operators in monte-carlo tree search. In *International conference on computers and games*, pages 72–83. Springer, 2006.

[35] Thomas M Cover and Joy A Thomas. *Elements of information theory.* John Wiley & Sons, 2012.

[36] George B Dantzig. Discrete-variable extremum problems. *Operations research*, 5(2):266–288, 1957.

[37] Adriaan D De Groot, Fernand Gobet, and Riekent W Jongman. *Perception and memory in chess: Studies in the heuristics of the professional eye.* Van Gorcum & Co, 1996.

[38] Thomas G Dietterich. State abstraction in MAXQ hierarchical reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 994–1000, 2000.

[39] Frank H Durgin, Brennan Klein, Ariana Spiegel, Cassandra J Strawser, and Morgan Williams. The social psychology of perception experiments: Hills, backpacks, glucose, and the problem of generalizability. *Journal of Experimental Psychology: Human Perception and Performance*, 38(6):1582, 2012.

[40] Ward Edwards. The theory of decision making. *Psychological bulletin*, 51(4):380, 1954.

[41] David Enoch and Joshua Schechter. How are basic belief-forming methods justified? *Philosophy and Phenomenological Research*, 76(3):547–579, 2008.

[42] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018.

[43] Chaz Firestone and Brian J Scholl. Cognition does not affect perception: Evaluating the evidence for top-down effects. *Behavioral and Brain Sciences*, 39, 2016.

[44] Jerry A Fodor. *The language of thought*, volume 5. Harvard University Press, 1975.

[45] Daniel Garber. Old evidence and logical omniscience in bayesian confirmation theory. 1983.

[46] Peter Gärdenfors. Induction, conceptual spaces and ai. *Philosophy of Science*, 57(1):78–95, 1990.

[47] Peter Gärdenfors. *Conceptual spaces: The geometry of thought.* MIT press, 2004.

[48] Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58, 1992.

[49] Samuel J Gershman, Eric J Horvitz, and Joshua B Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245): 273–278, 2015.

[50] Gerd Gigerenzer. *Adaptive thinking: Rationality in the real world.* Oxford University Press, USA, 2000.

[51] Gerd Gigerenzer and Daniel G Goldstein. Reasoning the fast and frugal way: models of bounded rationality. *Psychological review*, 103(4):650, 1996.

[52] Gerd Gigerenzer and Peter M Todd. Ecological rationality: the normative study of heuristics. In *Ecological rationality: Intelligence in the world*, pages 487–497. Oxford University Press, 2012.

[53] John C Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 148–177, 1979.

[54] Alison Gopnik, Andrew N Meltzoff, and Peter Bryant. *Words, thoughts, and theories*, volume 1. MIT Press Cambridge, MA, 1997.

[55] Alison Gopnik, Andrew N Meltzoff, and Patricia Katherine Kuhl. *The scientist in the crib: What early learning tells us about the mind.* Perennial New York, NY, 2001.

[56] Thomas L Griffiths, Falk Lieder, and Noah D Goodman. Rational use of cognitive resources: Levels of analysis between the computational and the algorithmic. *Topics in Cognitive Science*, 7(2):217–229, 2015.

[57] Sven Ove Hansson. Decision Theory: A Brief Introduction. *Department of Philosophy and the History of Technology. Royal Institute of Technology. Stockholm*, 1994.

[58] Jean Harb, Pierre-Luc Bacon, Martin Klissarov, and Doina Precup. When waiting is not an option: Learning options with a deliberation cost. *arXiv preprint arXiv:1709.04571*, 2017.

[59] Jean Harb, Pierre-Luc Bacon, Martin Klissarov, and Doina Precup. When waiting is not an option: Learning options with a deliberation cost. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[60] Gilbert Harman and Sanjeev Kulkarni. *Reliable reasoning: Induction and statistical learning theory.* MIT Press, 2012.

[61] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4 (2):100–107, 1968.

[62] Eric Horvitz. Reasoning under varying and uncertain resource constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 88, pages 111–116, 1988.

[63] Eric Horvitz and Shlomo Zilberstein. Computational tradeoffs under bounded resources. *Artificial Intelligence*, 126(1-2):1–4, 2001.

[64] Eric J Horvitz. Reasoning about beliefs and actions under computational resource constraints. In *Workshop on Uncertainty in Artificial Intelligence*, 1987.

[65] David Hume. *A Treatise of Human Nature*. Courier Corporation, 2003.

[66] Glenn A Iba. A heuristic approach to the discovery of macro-operators. *Machine Learning*, 3(4):285–317, 1989.

[67] Mark Jago. Hintikka and cresswell on logical omniscience. *Logic and Logical Philosophy*, 15 (4):325–354, 2007.

[68] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

[69] Nan Jiang, Alex Kulesza, and Satinder Singh. Abstraction selection in model-based reinforcement learning. In *International Conference on Machine Learning*, pages 179–188, 2015.

[70] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713, 2017.

[71] David S Johnson. Approximation algorithms for combinatorial problems. *Journal of computer and system sciences*, 9(3):256–278, 1974.

[72] Eric J Johnson and Daniel Goldstein. Do defaults save lives?, 2003.

[73] Nicholas K Jong and Peter Stone. State abstraction discovery from irrelevant state variables. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 752–757, 2005.

[74] Anders Jonsson and Andrew G Barto. Automated state abstraction for options using the U-tree algorithm. In *Advances in Neural Information Processing Systems*, pages 1054–1060, 2001.

[75] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[76] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1):99–134, 1998.

[77] Daniel Kahneman and Patrick Egan. *Thinking, fast and slow*, volume 1. Farrar, Straus and Giroux New York, 2011.

[78] Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, pages 99–127. World Scientific, 2013.

[79] Sham Machandranath Kakade et al. *On the sample complexity of reinforcement learning.* PhD thesis, University of London, 2003.

[80] Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

[81] Michael J Kearns, Umesh Virkumar Vazirani, and Umesh Vazirani. *An Introduction To Computational Learning Theory.* MIT press, 1994.

[82] Frank H Knight. *Risk, uncertainty and profit.* Courier Corporation, 2012.

[83] Varun Raj Kompella, Marijn Stollenga, Matthew Luciw, and Juergen Schmidhuber. Contin-

ual curiosity-driven skill acquisition from high-dimensional video inputs for humanoid robots. *Artificial Intelligence*, 247:313–335, 2017.

[84] George Konidaris and Andrew Barto. Skill discovery in continuous reinforcement learning domains using skill chaining. In *Advances in Neural Information Processing Systems*, pages 1015–1023, 2009.

[85] George Konidaris, Leslie Pack Kaelbling, and Tomas Lozano-Perez. From skills to symbols: Learning symbolic representations for abstract high-level planning. *Journal of Artificial Intelligence Research*, 61:215–289, 2018.

[86] James J Kuffner and Steven M LaValle. Rrt-connect: An efficient approach to single-query path planning. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 2, pages 995–1001. IEEE, 2000.

[87] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[88] Steven M LaValle. Rapidly-exploring random trees: A new tool for path planning. 1998.

[89] Steven M LaValle. *Planning Algorithms*. Cambridge University Press, 2006.

[90] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436, 2015.

[91] Leonid A Levin. Average case complete problems. *SIAM Journal on Computing*, 15(1): 285–286, 1986.

[92] Richard L Lewis, Andrew Howes, and Satinder Singh. Computational rationality: Linking

mechanism and behavior through bounded utility maximization. *Topics in cognitive science*, 6(2):279–311, 2014.

[93] Lihong Li, Thomas J Walsh, and Michael L Littman. Towards a unified theory of state abstraction for mdps. In *ISAIM*, 2006.

[94] Lihong Li, Michael L Littman, and Thomas J Walsh. Knows what it knows: a framework for self-aware learning. In *Proceedings of the International Conference on Machine learning*, pages 568–575, 2008.

[95] Zhi Li and Frank H Durgin. Perceived slant of binocularly viewed large-scale surfaces: A common model from explicit and implicit measures. *Journal of Vision*, 10(14):13–13, 2010.

[96] Falk Lieder, Thomas L Griffiths, Quentin JM Huys, and Noah D Goodman. Empirical evidence for resource-rational anchoring and adjustment. *Psychonomic Bulletin & Review*, 25(2):775–784, 2018.

[97] Michael L Littman, Thomas L Dean, and Leslie Pack Kaelbling. On the complexity of solving Markov decision problems. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 394–402. Morgan Kaufmann Publishers Inc., 1995.

[98] Graham Loomes and Robert Sugden. Regret theory: An alternative theory of rational choice under uncertainty. *The Economic Journal*, 92(368):805–824, 1982.

[99] Marlos C Machado and Michael Bowling. Learning purposeful behaviour in the absence of rewards. *arXiv preprint arXiv:1605.07700*, 2016.

[100] Marlos C Machado, Marc G Bellemare, and Michael Bowling. A Laplacian framework for

option discovery in reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2017.

[101] Marlos C Machado, Clemens Rosenbaum, Xiaoxiao Guo, Miao Liu, Gerald Tesauro, and Murray Campbell. Eigenoption discovery through the deep successor representation. *arXiv preprint arXiv:1710.11089*, 2017.

[102] Odalric-Ambrym Maillard, Timothy A Mann, and Shie Mannor. How hard is my mdp?" the distribution-norm to the rescue". In *Advances in Neural Information Processing Systems*, pages 1835–1843, 2014.

[103] Sultan Javed Majeed and Marcus Hutter. Performance guarantees for homomorphisms beyond Markov decision processes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[104] David A McAllester. Some PAC-Bayesian Theorems. *Machine Learning*, 37(3):355–363, 1999.

[105] Amy McGovern and Andrew G Barto. Automatic discovery of subgoals in reinforcement learning using diverse density. In *Proceedings of the International Conference on Machine Learning*, 2001.

[106] Ishai Menache, Shie Mannor, and Nahum Shimkin. Q-cut - dynamic discovery of sub-goals in reinforcement learning. In *European Conference on Machine Learning*, pages 295–306. Springer, 2002.

[107] Tom M Mitchell. *The need for biases in learning generalizations.* Department of Computer Science, Laboratory for Computer Science Research , 1980.

[108] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[109] Oskar Morgenstern and John Von Neumann. *Theory of games and economic behavior.* Princeton university press, 1953.

[110] Robin Morris and Geoff Ward. *The cognitive psychology of planning.* Psychology Press, 2004.

[111] Brendan Mumey and Tomáš Gedeon. Optimal mutual information quantization is NP-complete. In *Neural Information Coding*, 2003.

[112] Gregory Murphy. *The big book of concepts.* MIT press, 2004.

[113] Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. In *Advances in Neural Information Processing Systems*, pages 9209–9220, 2018.

[114] Allen Newell, John C Shaw, and Herbert A Simon. Report on a general problem solving program. In *IFIP congress*, volume 256, page 64. Pittsburgh, PA, 1959.

[115] Yael Niv. Reinforcement learning in the brain. *Journal of Mathematical Psychology*, 53(3): 139–154, 2009.

[116] Daniel Alexander Ortega and Pedro Alejandro Braun. Information, utility and bounded rationality. In *International Conference on Artificial General Intelligence*, pages 269–274. Springer, 2011.

[117] Pedro Alejandro Ortega Jr. *A unified framework for resource-bounded autonomous agents interacting with unknown environments.* PhD thesis, University of Cambridge, 2011.

[118] P Ortner and R Auer. Logarithmic online regret bounds for undiscounted reinforcement learning. In *Proceedings of the 2006 Conference on Advances in Neural Information Processing Systems*, volume 19, page 49, 2007.

[119] Ian Osband and Benjamin Van Roy. On lower bounds for regret in reinforcement learning. *arXiv preprint arXiv:1608.02732*, 2016.

[120] Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. *Proceedings of the International Conference on Machine Learning*, 2014.

[121] Silviu Pitis. Rethinking the discount factor in reinforcement learning: A decision theoretic approach. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[122] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming.* John Wiley & Sons, 2014.

[123] Balaraman Ravindran. *SMDP homomorphisms: An algebraic approach to abstraction in semi Markov decision processes.* PhD thesis, University of Massachusetts Amherst, 2003.

[124] Ronald A Rensink, J Kevin O'Regan, and James J Clark. To see or not to see: The need for attention to perceive changes in scenes. *Psychological Science*, 8(5):368–373, 1997.

[125] Martin Riedmiller, Roland Hafner, Thomas Lampe, Michael Neunert, Jonas Degrave, Tom van de Wiele, Vlad Mnih, Nicolas Heess, and Jost Tobias Springenberg. Learning by playing–solving sparse reward tasks from scratch. In *Proceedings of the International Conference on Machine Learning*, volume 80, pages 4344–4353, 2018.

[126] Eleanor Rosch and Carolyn B Mervis. Family resemblances: Studies in the internal structure of categories. *Cognitive psychology*, 7(4):573–605, 1975.

[127] Jonathan Rubin, Ohad Shamir, and Naftali Tishby. Trading value and information in mdps. In *Decision Making with Imperfect Decision Makers*, pages 57–74. Springer, 2012.

[128] Stuart Russell and Peter Norvig. A modern approach. *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, 25:27, 1995.

[129] Stuart Russell and Devika Subramanian. Provably Bounded-Optimal Agents. *Journal of Artificial Intelligence Research*, 2:575–609, 1995. ISSN 1076-9757.

[130] Stuart Russell and Eric Wefald. Principles of metareasoning. *Artificial Intelligence*, 49(1-3): 361–395, 1991.

[131] Wolfram Schultz, Peter Dayan, and P Read Montague. A neural substrate of prediction and reward. *Science*, 275(5306):1593–1599, 1997.

[132] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

[133] Claude E Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 1948.

[134] Roger N Shepard. Stimulus and response generalization: A stochastic model relating generalization to distance in psychological space. *Psychometrika*, 22(4):325–345, 1957.

[135] Roger N Shepard. Toward a universal law of generalization for psychological science. *Science*, 237(4820):1317–1323, 1987.

[136] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529 (7587):484, 2016.

[137] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[138] Herbert A Simon. Theories of bounded rationality. *Decision and organization*, 1(1):161–176, 1972.

[139] Chris R Sims. Rate–distortion theory and human perception. *Cognition*, 152:181–198, 2016.

[140] Chris R Sims. Efficient coding explains the universal law of generalization in human perception. *Science*, 360(6389):652–656, 2018.

[141] Ö. Şimşek and A.G. Barto. Using relative novelty to identify useful temporal abstractions in reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, pages 751–758, 2004.

[142] O. Simsek, A.P. Wolfe, and A.G. Barto. Identifying useful subgoals in reinforcement learning by local graph partitioning. In *Proceedings of the International Conference on Machine Learning*, pages 816–823, 2005.

[143] Özgür Şimşek and Andrew G Barto. Skill characterization based on betweenness. In *Advances in Neural Information Processing Systems*, pages 1497–1504, 2009.

[144] Michael Sipser. *Introduction to the Theory of Computation*, volume 2. Thomson Course Technology Boston, 2006.

[145] Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial life*, 11(1-2):13–29, 2005.

[146] Vernon L Smith. Constructivist and ecological rationality in economics. *American economic review*, 93(3):465–508, 2003.

[147] Declan Smithies. Ideal rationality and logical omniscience. *Synthese*, 192(9):2769–2793, 2015.

[148] Declan Smithies and Daniel Stoljar. *Introspection and consciousness.* Oxford University Press, 2012.

[149] Katie Steele and H. Orri Stefnsson. Decision theory. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy.* Metaphysics Research Lab, Stanford University, winter 2016 edition, 2016.

[150] Martin Stolle and Doina Precup. Learning options in reinforcement learning. In *International Symposium on abstraction, reformulation, and approximation*, pages 212–223. Springer, 2002.

[151] Alexander L Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L Littman. Pac model-free reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 881–888. ACM, 2006.

[152] Alexander L Strehl, Lihong Li, and Michael L Littman. Reinforcement learning in finite MDPs: PAC analysis. *Journal of Machine Learning Research*, 10(Nov):2413–2444, 2009.

[153] Tobias Sutter, David Sutter, Peyman Mohajerin Esfahani, and John Lygeros. Efficient Ap-

proximation of Channel Capacities. *IEEE Transactions on Information Theory*, 61:1649–1666, 2015.

[154] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[155] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction.* MIT press Cambridge, 2nd edition, 2018.

[156] Richard S Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1): 181–211, 1999.

[157] István Szita and Csaba Szepesvári. Model-based reinforcement learning with nearly tight exploration complexity bounds. In *Proceedings of the International Conference on Machine Learning*, pages 1031–1038, 2010.

[158] Adrien Ali Taïga, Aaron Courville, and Marc G Bellemare. Approximate exploration through state abstraction. *arXiv preprint arXiv:1808.09819*, 2018.

[159] Jonathan Taylor, Doina Precup, and Prakash Panagaden. Bounding performance loss in approximate MDP homomorphisms. In *NeurIPS*, 2008.

[160] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[161] Naftali Tishby and Daniel Polani. Information theory of decisions and actions. In *Perception-action cycle*, pages 601–636. Springer, 2011.

[162] Naftali Tishby, Fernando C Pereira, and William Bialek. The Information Bottleneck Method. *The 37th Annual Allerton Conference on Communication, Control, and Computing*, 1999.

[163] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[164] Michael Titelbaum. Fundamentals of bayesian epistemology, 2015.

[165] Peter M Todd and Gerd Gigerenzer. Environments that make us smart: Ecological rationality. *Current directions in psychological science*, 16(3):167–171, 2007.

[166] Peter M Todd and Gerd Gigerenzer. *Ecological rationality: Intelligence in the world.* Oxford University Press, 2012.

[167] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974.

[168] Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[169] Vladimir N Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer, 2015.

[170] Vladimir Naumovich Vapnik and Vlamimir Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.

[171] Vijay V Vazirani. *Approximation algorithms.* Springer Science & Business Media, 2013.

[172] John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior (commemorative edition)*. Princeton University Press, 2007.

[173] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[174] Deirdre Wilson and Dan Sperber. Relevance theory. In *Handbook of pragmatics*. Blackwell, 2002.

[175] David H Wolpert, William G Macready, et al. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[176] Shlomo Zilberstein. Models of bounded rationality–a concept paper. 1995.

[177] Shlomo Zilberstein. Operational rationality through compilation of anytime algorithms. *AI Magazine*, 16(2):79–79, 1995.

[178] Shlomo Zilberstein. Metareasoning and bounded rationality. *Metareasoning: Thinking about Thinking, MIT Press*, 2008.