A Digital Theory of Knowledge

Kristian Rönn
Advisor: Prof. Erik Olsson
FTEK01, Spring 2012
Department of Philosophy,
Lund University

June 27, 2012

Abstract

The goal with this paper is to formally define knowledge from the assumption that our universe is computable. Based on this assumption, we will formulate a minimalist ontology that will be the theoretical basis for our formal definition of knowledge. We will use the classical definitions of knowledge like "reliabilism" (RTB) and "justified true belief" (JTB) as our starting point and formally translate them to our digital framework. To do this we will investigate what it means for a process to be as reliable as possible, in a theoretical sense, by presenting Ray Solomonoff's induction as a solution to the shortcomings with Baysian inference. We will then criticize JTB and RTB and in the spirit of Rudolf Carnap's idea of explications stipulate two, mutually inclusive and complementary definitions of knowledge that we will call "generative knowledge" and "absolute knowledge".

Contents

1 Introduction

The idea that formal logic can be useful for reasoning about philosophy is very much at the core of analytical philosophy. Computer science has for the past decades changed much of science with he introduction of computational methods in applied fields like physics, chemistry, economics and biology. Not many people know that computer science is, in a way, the child of Frege's philosophy and arguably the greatest achievement of philosophy in general. Bertrand Russell and Alfred North Whitehead continued Frege's work in their magnum opus Principa Matematica, after Russell discovered a logical paradox in Frege's work. This in turn motivated Gödel to formulate his incompleteness theorem in 1931, which demonstrated that Russell and Whitehead's logical system was incomplete. Alan Turing then generalized Kurt Gödel's results in 1936 by replacing Gödel's universal arithmetic-based formal language with what became known as Turing machines. Thus the field of computer science was born, and it all started by a philosophical search for the foundations of mathematics!

Since that time, philosophy and computer science have drifted apart. Digital philosophy is an attempt to unite the foundations of physics, computer science, mathematics and philosophy under a discrete digital framework. Konrad Zuse, the man behind the world's first functional program-controlled Turing-complete computer was also the great pioneer of digital philosophy. Zuse argued in his paper "Rechnender Raum" (1969)[24] that the universe was a digital computer. Edward Fredkin was the first physicist to embrace the idea in the 70s and he coined the term "digital philosophy". The idea was revived again in the middle of the 90s and are being developed mainly by physicists and computer scientists like Stephen Wolfram[23], David Deusch[6], Marcus Hutter[15], Jürgen Schmidhuber[17], Max Tegmark[20], Gregory Chaitin[4], Gerard 't Hooft[19] and Seth Llyde[12]. But unfortunately have not yet had any impact among philosophers.

The goal with this paper is to formally define knowledge from a digital philosophical framework with as few philosophical assumptions as possible. The fundamental assumptions are:

1. The empirical assumption that we get knowledge by observing the outside world. (section 1.1).

2. The digital philosophical assumption that all that has physical existence can be described by a computer program (section 2) which implies determinism and a subjectivist interpretation of probability (section 1.2).

Based on these assumptions, we will formulate a minimalist ontology (section 3) that will be the theoretical basis for our formal definition of knowledge (section 4). We will use the classical definitions of knowledge like "reliabilism" (RTB) and "justified true belief" (JTB) as our starting point and formally translate them to our digital framework. To do this we will investigate what it means for a process to be as reliable as possible, in a theoretical sense, by presenting Ray Solomonoff's induction as a solution to the shortcomings with Baysian inference (section 4.1-4.2). We will then criticize JTB and RTB and in the spirit of Rudolf Carnap's idea of explications stipulate two, not mutually exclusive but complementary, definitions of knowledge that we will call "generative knowledge" and "absolute knowledge" (section 4.3).

1.1 Empiricism vs. Rationalism

Because empiricism is one of the fundamental assumptions in this paper, it is appropriate that we define what philosophical components empiricism contains. Table 1.1 below compares empiricism with rationalism.

The table is intended to provide an informative overview of dichotomies related to empiricism and rationalism. It is important to emphasize that the table is an oversimplification and that in reality it is a more complicated relationship between empiricism and rationalism. There is much to be said about the table below which I will not elaborate on in this paper. We will, for example, not talk about Kolomogorovs probability axioms. The reason for this is that we want the focus to be on the philosophical ideas and not on the technical details.

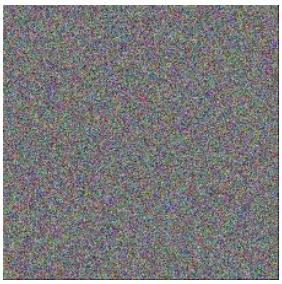

Figure 1: A visualization of the first 1.5 million digits of π seems indistinguishable from random noise.

	Rationalism	**Empiricism**
Knowledge	**A priori:** knowledge is independent of experience.	**A posteriori:** knowledge is dependent on experience.
Justification	**Consistency:** a statement is true if there are no contradictions in the reasoning from the premisis.	**Correspondence:** a hypothesis is true of it corresponds to the real state of the world.
Statements	**Analytical**: "bachelors are unmarried men"	**Synthetical**: "All bachelors are unhappy"
Reasoning	**Deductive:** valid inferences from a set of logical premises.	**Inductive:** "best" conclusion from a number of observations.
Academic field	Philosophy and Mathematics	Natural sciences (or all fields with a empirical method)
Logical system	**First-order logic:** **Start:** Logical axiomatic schemes **Input:** a statement X and domain-specific non-logical axioms **Method of inference:** modus ponens **Output:** $X \in \{0, 1\}$ where 1 (true) 0 (false)	**Bayesianism:** **Start:** Kolomogorovs probability axioms **Input:** prior probability for a hypotheses X. **Method of inference:** Bayes's theorem **Output:** posterior probability for $X \in [0, 1]$ where P (X) = 1 (true) P (X) = 0 (false) 0 < P (X) < 1 (probable)

Table 1: Overview of the dichotomies related to empiricism and rationalism.

1.2 Philosophy of probability

Since inductive reasoning is based on probabilities, it may be appropriate to define the concept of probability. There are three main philosophical interpretations of what probability actually is [9]. A compact overview of the arguments for and against the three interpretations can be found in table 1.2 below. Of the

Frequentism	**Objectivism**	**Subjectivism**
If $k(n)$ the number of times E occurs in n trials, then $P(E) := lim_{n\to\infty}\frac{k(n)}{n}$. **Criticism:** no matter how large n we have, we can never know how far from the actual probability we are $\lvert k(n)/n - P(E)\rvert$. **Frequentist:** the probability of $k(n)/n$ to be close to $P(E)$ is larger for large n. **Counterargument:** The concept of probability is used in the answer above, resulting in circularity because we assume what is to be defined.	Probability has objective existence and is embedded in nature. **Criticism:** nature is determined and therefore can not provide true randomness. **Objectivist:** It is far from clear that the universe is deterministic. Quantum mechanics seems to give proof to the contrary. **Counterargument:** In mathematics, deterministic rules can create a pseudo-random output. It would for example be impossible to distinguish 50 not already known digits of π from a "truly" random sequence[23, p.23-42] (see figure 1). It is therfore possible that even quantum phenomena is deterministic. [19] To refer to a mysterious and non-analyzable concept as randomness is superfluous (and perhaps even unscientific [17, p. 4]).	Probability is the degree of belief an agent has that an event will occur. **Criticism:** agents with different histories will ascribe different probabilities to the same event. Example: A flips a coin that ends up on heads 745/1000 times.$\Rightarrow$ $P(heads) = 0.745$. B flips the same coin and it lands on heads 431/1000 times. $\Rightarrow$ $P(heads) = 0.431$. **Subjectivist:** What side the coin lands on is determined by the laws of physics. To attribute the coin an "objective" probability of $P(heads) = 0.5$ is therfore incorrect. A's and B's confidence will also converge if they observe the same source and update their beliefs based on new data in a rational way.

Table 2: Overview of the main philosophies of what probabilities are.

three positions, subjectivism seems to be the most reasonable one in the sense that it makes the fewest assumptions, is compatible with determinism and implies that probability can be understood scientifically. In this paper we will therefore adopt a subjectivist approach. The comparison above is once again an oversimplification and does not include things like Richard von Mises "Axiom of Convergence" on the frequentist side of the argument[**?**].

2 Theoretical Foundations

The goal of this section is to present the theoretical framework that the rest of this paper is built upon. According to the subjectivist position , there exists no true randomness, and everything follows well defined rules. So the first thing we must do is to formalize exactly what is meant by a rule.

2.1 Turing machines

Formally, a rule can be viewed as an algorithm executed by a computer. An algorithm is simply a number of atomic instructions that tell the computer what to do. Algorithms can easiest be illustrated with flowcharts. One of the most famous and earliest examples of an algorithm is the Euclidean algorithm that tells you how to compute the lowest common denominator of two numbers (see figure 2).

A recipe for sponge cake could informally also be seen as an algorithm. Just as it can take differing amounts of time and difficulty for different people to bake a cake with the cake algorithm, it takes different lengths of time for different computers and programming languages to perform the atomic steps in an algorithm / computer program. For this reason, theoretical computer scientists use a theoretical computer model called a Turing machine.

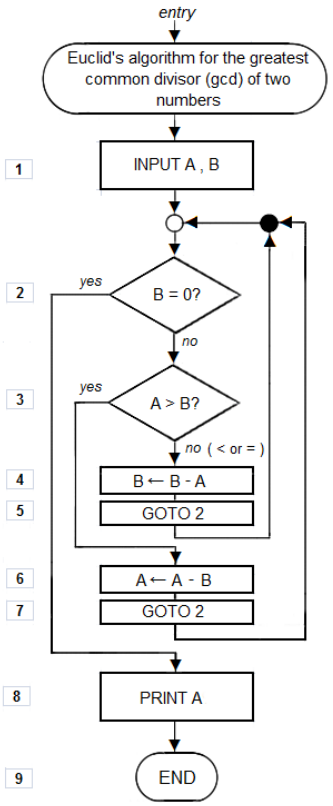

Figure 2: Euclid's algorithm

Definition 1. *A Turing machine T is a machine that outputs a string of symbols x when it is executed.*

Definition 2. *$U(v) = x$ denotes a universal Turing machine U which outputs a string of symbols x when it executes the program v.*

The details of how a Turing machine functions is not relevant for this paper and the definitions above are slightly simplified. The main difference between a Turing machine and a Universal Turing machine is that the universal Turing machine is programmable in the sense that its output depends on the program it runs, while a non-universal Turing machine always runs the same program. If a non-universal Turing machine is an elevator that always does the same thing when you press keys (ie go to the floor you press), then we can think of a universal Turing machine as a programmable personal computer that can do everything (even behave as an elevator), depending on how you program it.

The interesting question is whether there are things that a Turing machine can not do.

The Church- Turing Thesis: "Everything that is algorithmically computable can be computed by a Turing machine".

A common misconception is that the Church-Turing thesis shows that Turing machines can compute the solution to any problem that can be solved by instructions, explicitly stated rules, or procedures. This is simply not true. The Church-Turing thesis should instead of a thesis rather be seen as a stimulative definition of what is algorithmically computable.

There are also several so-called uncomputable problems that can not be solved by a Turing machine (such as the halting problem)[5]. But the interesting question for our analysis is whether there are phenomena in the universe that can not be calculated and described by a computer.

Church- Turing- Deutsch principle: "There are no physical phenomena that can not be computed by a Turing machine."[6, p.123-141]

Church-Turing-Deutsch principle which was formulated in 1985 by David Deutsch implies that our universe is deterministic because it must follow strict rules to be described by a Turing machine. This can partly be seen as a formalization of the subjectivist thesis.

2.2 Kolmogorov complexity

Another important concept for our future analysis is the Kolmogorov complexity, or Kolmogorov- Chaitin complexity as it is sometimes called, of a string x denoted $K(x)$.

Definition 3. $K_U(x) := min_p \{len(p) : U(p) = x\}$

The Kolomorogov complexity of a string x is thus the length of the shortest program that gives x as an output when executed by the universal Turing machine U.[4, p.107-124][11, p.31]
Example: Let $x = 1111...$ be a string of 1000 1s.

- Let U_N be a universal Turing machine that is built to manage our natural language. The shortest program on U_N that gives x as output can be printed "1000 one's" which is a string of seven symbols$\Rightarrow K_{U_N}(x) = 10$.
- Let U_J be a universal Turing machine that is built to handle Java. The shortest program on U_J that gives x as the output can be written

  `"for(i=0;i<1000;i++) print(“1”);"`

  which is a string of 30 symbols $\Rightarrow K_{U_J}(x) = 30$

The above example also illustrates the problem that the Kolomorogov complexity of the string will be different depending on the Turing machine we use as a reference. For every complex string x that is measured with U as the reference machine there exists another reference machine U 'such that $K_{U'}(x) = 1$.To solve this problem we introduce the concept of Natural Turing machines.[15, p.45-46]

Definition 4. *A universal Turing machine is considered natural if it does not contain any extreme biases. In other words, if it does not make any arbitrary, intuitively complex strings appear simple.*

As of now U will refer to a natural Turing machine. A good theoretical property of Kolmogorov complexity is that it generally is the same regardless of the Universal Turing machine is chosen. This is because the universal Turing machines can simulate each other with an extra constant input bits in the same way that a computer can simulate an elevator if we program it to do so.[11, p.32]

3 Digital Ontology

In this section we will define what an agent is, what a world is and what truth is from the theoretical framework in the previous section. Most of the notation is borrowed from Marcus Hutter[15].

In this digital ontological framework every observation x by an agent is produced by the natural Turing machine U when running an program v, called the agents environment.

3.1 The World

An arbitrary world W can be defined as a tuple $W = (U, X, M_U)$.

Definition 5. *The observation space X is the finite set of atomic objects that can exist in a universe. In theoretical computer science, it is called an alphabet and is the set of symbols the Turing machine output may consist of.*

Theorem 1. *For all X , $x \in X$ can be encoded to a binary string in the alphabet $\{0,1\}$. We can therefore from now on, let $X = \{0,1\}$ without losing any expressive power.*

- **Proof**: Every symbol can be encoded as a string of 1:s, where 0:s are used to separate different observations. In an environment where there is nothing more than a dice thrown $X = \{1,2,3,4,5,6\}$ can be encoded to $X = \{1, 11, 111, 1111, 11111, 111111\}$. 11011101 is an encoding of the outcomes 2,3,1.
- X^* is the set of all finite histories a universe can have. $*$ is an operation that creates a new set X^{ast} which consists of all enumerable combinations of an alphabet .
- X^∞ is the set of all infinite histories a universe can have. ∞ is an operation that creates a new set X^∞ which consists of all non-enumerable combinations of an alphabet .
- **Example:** If the alphabet is $X = \{0,1,2,3,4,5,6,7,8,9,.\}$ then $X^* = \mathbb{Q}$ and $X^\infty = \mathbb{R}$ where '.' is the decimal point. $\mathbb{Q}$ is enumerable while $\mathbb{R}$ is not, as we know from Cantor's diagonal argument.

Definition 6. *The Enviroment class M_U is the set of all environmental programs v.*

Theorem 2. *All $v \in M_U$ can with so-called Gödel Encoding be encoded into natural numbers which, in turn, can be converted into binary numbers. Thus, we can let $M_U = \{0,1\}^*$ without losing expressive power[7, p.25-26]. This also means that M_U is enumerable*

Since M_U is the set of all programs, it will contain, among other things, the source code for Windows 7 and Tetris as well as the perfect description of our universe and every other conceivable universe (because we assume that the Church-Turing-Deutsch Principle is true). Some scientists, most notably Tegmark[20] and Schmidthuber[17] propose that M_U have real physical existence as a multiverse. In other words, every possible program has physical existence and are therefore more than just abstractions. This view is sometimes called constructivist quasi-empiricism in the philosophy of mathematics.

Definition 7. *It is possible to identify at least three different 'sizes' of environments (see figure 3).*

- *$q \in M_U$ is a "theory of everything" (TOE):$\Leftrightarrow u^q := U(q) \in X^\infty$ is the entire history of the universe generated by the Turing machine U when running q. u could be seen as a long 3D movie where all episodes from the Big Bang to the destruction or infinite continuation of the universe is included.[8]*
- *$(q,s) \in M_U$ is an agent environment $:\Leftrightarrow o^{qs} := U(q,s) \in X^*$ is the agent's observation history where $s \in M_U$ is an observer program that extracts the observations from the ToE $q \in M_U$. s could be seen as a 3D camera that is programmed to go around and record different parts of the universe a finite time and the result is the movie o^{qs}.[8]*
- *(o^{qs}, p) is a partial environment $:\Leftrightarrow x^{qsp} := U(x^{qs}, s) \in X^*$ is the agent's observation of an object or episode, where $p \in M_U$ is a program that extracts the object or event x^{qs} from the agents history $x^{qs} \in M_U$. s could for example be an image recognition algorithm that distinguishes foreground object from the background in images from different angles.*

Marcus Hutter points out that one must include the observere program s in a complete ToE to make any meaningful predictions[8, p.9-12]. If our ToE does not tell us that we are an observer who find ourselves on Earth rather than somewhere in the Andromeda, we can not make any predictions about what we should expect to see when we look up at the night sky.

Definition 8.

- *An environment q is fully observable for an agent $s :\Leftrightarrow x^{qs} = u$*
- *An environment q is partially observable to an agent $s :\Leftrightarrow x^{qs} \neq u$*

The definitions above correspond well with our intuition of what it means that something is fully observable. For a chess computer, its environment is fully observable because it sees everything that happens on the board. Our environment is to us, however, only partially observable because we can not possibly see everything happening simultaneously throughout the universe.

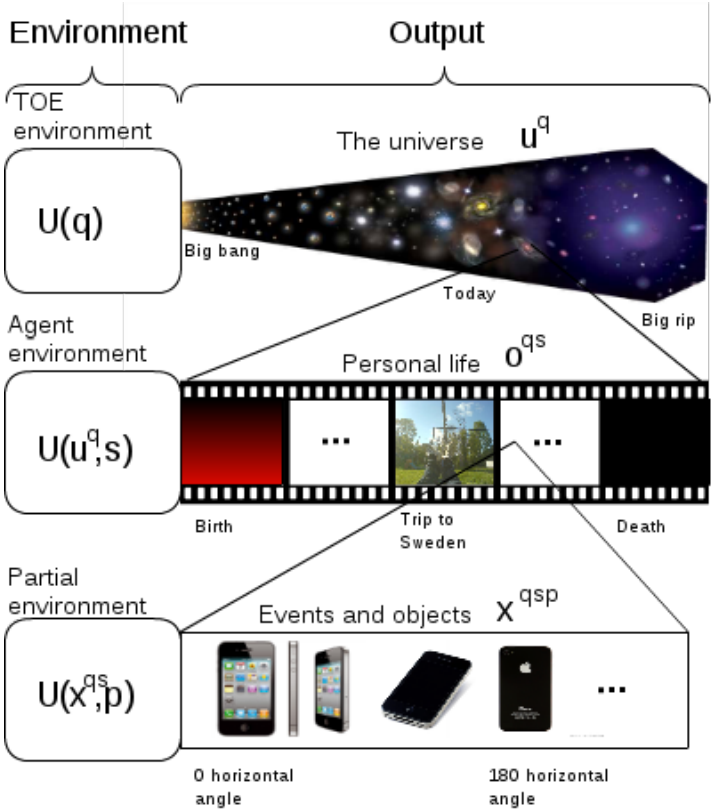

Figure 3: Three different 'sizes' of environments.

If the environment is fully observable, then metaphysical solipsism can be regarded as true in the sense that the agent's sensations constitute the entire universe.

One possible way to determine whether one should believe that an external reality exists outside oneself would be to check if o^{qs} can be compressed more when it is seen as the result of a partially observable environment and an observer program (q_2, s) than if it is seen as a fully observable environment ToE q_1. If so, then realism would be justified by Occam's razor because it favours simple (short) theories that requires fewer bits to describe. Intuitively, it seems to be true in respect to our universe since it seems to require fewer bits of information to describe a theory where the sun rises and sets regularly, whether I look at it or not. Occam's razor will be discussed more and formally justified in section 4.2.

3.2 The Agent

An arbitrary agent A can be defined as a tuple $A = (P, [0,1], X, M)$. It is important to note that we will only analyse passive agents, who only observe but do not interact with their environment, in this paper.

Let $x \in X^*$ be a arbitrary substring of the agents entire observation history

o^{qs} and let $x_{m:n} \in X^*$ be a substring of x that contains all the bits $x_m ... x_n$ where x_i denotes the i:th bit of x

Definition 9. *The hypothesis class $M \subseteq M_U$ is the set of all hypotheses H_v the agent has about what environment that produces x.*

- **Example**: Physicists are discussing whether string theory will be included in M or not.

Definition 10. *A hypothesis $H_v \in M$ is true $:\Leftrightarrow U(H_v) = x \in X^*$. Let μ denote the true hypothesis.*

A hypothesis H_v is said to be true if and only if it is a model of the real environment that gives the observation x as an output. This can bee seen as a formalization of the correspondence criteria of truth, where a hypothesis is true if it corresponds to the real environment. In other words: the map corresponds to the territory.

Definition 11. *The inference system P is a function $P : X \to M \to [0, 1]$ that assigns a probability distribution over the agents hypothesis class M given an observation x.*

A natural question is what P that counts as a 'reliable" inference system (ie. what probability distribution over what hypothesis class should the agent have given the observations). We will examine that question in the next section and conclude that Solomonoff induction is, in theory, the optimal system of inference.

- **Example**: Let our environment be the flipping of a coin where the only two observations are $X = \{Heads, Tails\}$ and let the hypothesis class $M = \{H_{Heads}\}$ where H_{Heads} is the hypothesis that the coin will land on heads the next time. A possible inference system for the agent could be $P(H_{Heads}) = \frac{h}{n}$ where n is number of times we have tossed the coin and h is the number of times the coin landed on heads.
- $P(H_v)$ is our *prior* belief that $\mu = H_v \in M$, ie. our original belief that one of our hypothesis H_v is true.
- $P(x|H_v)$ is the probability of x given H_v, i.e. probability that the environment H_v will produce x.

Definition 12.

- *The environment H_v is* ***deterministic*** $:\Leftrightarrow P(x|H_v) \in \{1, 0\}$.
- *The environment H_v is* ***stochastic*** $:\Leftrightarrow P(x|H_v) \in [0, 1]$.

The definitions above are very intuitive and we can see that deterministic environments is a special case of stochastic environments. In the deterministic case an environment H_v is determined to either produce an observation ($P(x|H_v) = 1$) or not ($P(x|H_v) = 0$). In the stochastic case then it is chance that determines if an environmental H_v will give rise to an observation x or not.

Seemingly stochastic environments can, as we have said before, be explained deterministically by a pseudo-random output generated by the environmental

program. In fact, all beliefs in stochastic environments can be seen as a mix of deterministic environments. If I say that it will rain with 60% probability tomorrow, then what I am really saying is that I with 60% certainty believe that we live in an (deterministic) environment that is a member of the set of all environments that will generate a string with an episode of rain tomorrow.

4 Digital Epistemology

In this section we will use the digital ontological framework from the previous section to define what it means for an agent to have knowledge about the world.

We will focus on two of the most common approaches to defining knowledge called "reliabilism" (RTB) and "justified true belief" (JTB)[18]. We will not take sides in the externalism versus internalism debate and will therefore treat JTB and RTB as effectively equivalent in the rest of the paper. RTB and JTB are defined as follows:

Definition 13. *S has knowledge that p if and only if:[18]*

(i) p is true

(ii) S believes that p is true

(iii) S's belief that p is well founded (JTB)

(iii') S's belief that p was caused by a reliable process (RTB)

Our goal is to formalize this fuzzy definition of knowledge within the framework of digital philosophy to asses its validity. To be able to do that we must first define what it means for a process to be reliable or well founded.

In the next two subsection, we will introduce Bayesian inference from a digital ontological position, describe the problems with it, how Solomonoff induction solves these problems and how Solomonoff induction can be seen as the golden standard of reliable inductive inference systems.

4.1 Bayesian inference

The Bayesian method is used in many scientific fields in order to reason about probabilities, It consists of the following four steps.

1. Identify the possible hypotheses H_v and add them to your hypothesis class M of hypotheses you think might be true.

2. Decide which prior belief $P(H_v)$ you want to give each hypothesis in your hypothesis class M where $\sum_{H_i \in M} P(H_i) = 1$ and not more than one environment can be the real environment (i.e. the environments are mutually exclusive).

3. Use Bayes' theorem on each hypothesis $H_v \in M$ to update your **prior** belief for each hypothesis into a **posterior** belief based on the data $x_{1:n}$ you have observed.

4. Use hypotheses to make a prediction about the observation x_m you are expected to have next.

In this section we will see how to do steps 3-4. Much criticism has been directed against the Bayesian method because it does not have anything to say about 1-2. In the next section we will see how Solomonoff induction is building on the Bayesian framework by saying how to do step 1-2.

4.1.1 Step 4: Bayes' mixture

We start here to find a solution to Step 4 and work us down. Assume that we already have a probability distribution over a hypothesis class. How then do we decide which hypothesis we should use in our calculations of what the next observation will be? Let us reformulate the problem in a more intuitive way!

Pretend that every hypothesis in your hypothesis class is an expert who tries to predict the next symbol in the observation string. The question of which hypothesis to select can then be rephrased to which expert you should trust?

- **Alternative 1 (follow the leader):** Choose the expert that, after a certain number of times, performs the best. This solution may appear intuitive, but is, in fact, very misleading. It may be the case that the expert you chose just happened to be lucky in the first predictions and will make bad predictions in the future. To blindly trust a person just because he/she has been right so far is, thus, a poor strategy.
- **Alternative 2 (Bayes' mixture):** Randomize among the experts with a certain probability depending on how well they perform. This weighted summation xi is called" Bayes' mixture" or" marginal likelihood" and is defined as follows.

Definition 14. *The belief $\xi(x)$ for observing a string x given hypothesis class M is* $\xi(x) := \sum_{H_i \in M} P(x|H_i)P(H_i)$

Theorem 3. *An important mathematical property that the Bayes' mixture has is its dominance.* $\forall x \in X^* \forall H_v \in M : \xi(x) \geq P(H_v) \cdot P(x|H_v)$*[15, p.24]*

The theorem above says that an agent who uses the "follow the leader" strategy will always perform worse than an agent who uses mixture. This is very intuitive as" follow the leader" in the best case has the belief $P(\mu) = 1$ which results in μ being chosen all times also with Bayes mixture since the belief in the hypothesis class must sum to 1. We have, in other words, defined an agent's belief that a particular observation will occur from the belief in the hypotheses it is considering.

4.1.2 Step 3: Bayes' theorem

Definition 15. *$P(H_v \cap x_{1:n})$ is the probability that both the environment H_v and the observation $x_{1:n}$ are true.*

Definition 16. $P(H_v|x_{1:n}) := \frac{P(H_v \cap x_{1:n})}{P(x_{1:n})}$ *denotes the probability that H_v is true given that $x_{1:n}$ is true (i.e. observed).*

It is now possible to make the following deduction from the definitions above: $P(H_v \cap x_{1:n}) =_{(1)} P(x_{1:n} \cap H_v) \Leftrightarrow_{(2)} P(H_v|x_{1:n})P(x_{1:n}) = P(x_{1:n}|H_v)P(H_v) \Rightarrow$

Definition 17.

$$P(H_v|x_{1:n}) = \frac{P(x_{1:n}|H_v)P(H_v)}{P(x_{1:n})} =_{(3)} \frac{P(x_{1:n}|H_v)P(H_v)}{\sum_{H_i \in M} P(x_{1:n}|H_i)P(H_i)}$$

The formula is known as Bayes' theorem and describes what an agent should do in order to update its belief $P(H_v)$ in a hypothesis H_v when given some observation data x.

- $P(H_v)$ is as we have previously defined our prior belief that $\mu = H_v \in M$, i.e. the belief that our hypothesis is true.
- $P(H_v|x_{1:n})$ is our posterior belief that $H_v \in M = \mu$ after we observed $x_{1:n}$.
- $H_v(x_{1:n}) := P(x_{1:n}|H_v)$ is as we have previously defined the probability of $x_{1:n}$ given H_v, ie. the probability that H_v will give rise to $x_{1:n}$.

In the numerator, there is the probability that our observation is produced by precisely H_v and the denominator is the total probability that we would have observed data $P(x_{1:n})$. Intuitively, one can think of Bayes' theorem as "the probability of an observation given a hypothesis" divided by "the overall probability of the observation".

Some comments on the steps marked (1)-(3) in the deduction above may be needed.

(1) The probability that" it is raining and you read philosophy" is obviously the same as the probability of" you read philosophy and it's raining."

(2) We use definition 16 above.

(3) We use definition 14 above.

Bayesian inference within the framework of digital ontology can be most easily visualized with figure 4 on the next page.

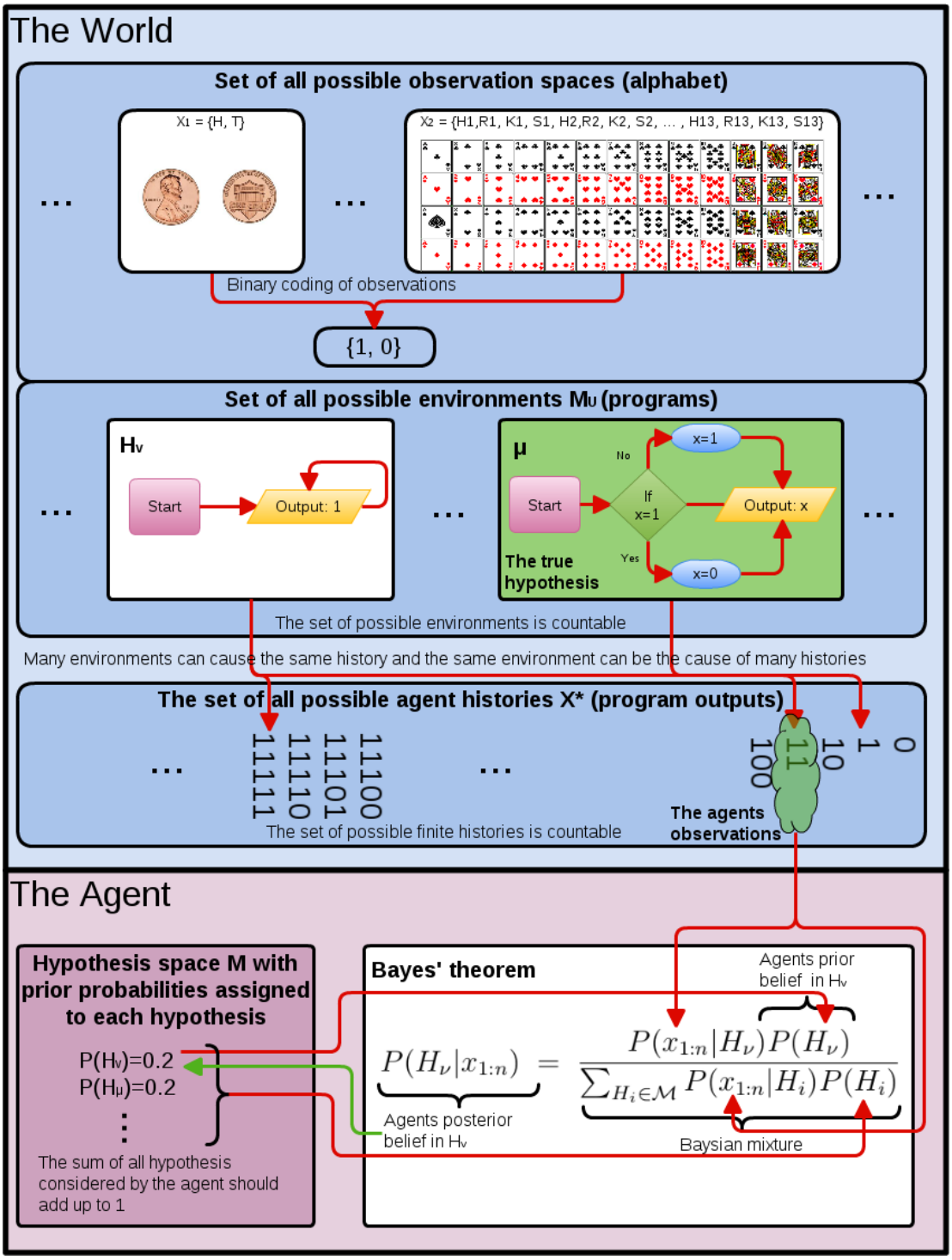

Figure 4: Visual overview of Bayesian inference.

4.2 Solomonoff induction

In this section we'll go over how Solomonoff induction explains how the steps 1-2 in the Bayesian method should be implemented in the best possible way and how Solomonoff induction can be seen as the gold standard for all inductive systems.

1. How do we choose which hypothesis H_v we should put in hypothesis class M?

2. How do we choose our prior probability distribution P over the hypothesis in class M?

Karl Popper's Falsifiability Principle is frequently invoked when it comes to 1. But it seems to be unhelpful in our framework because all hypotheses in M, by definition, are falsifiable since they tell us what observations that are expected to be seen.

Concerning 2, it is common to use the indifference principle that says you should give all the hypotheses a uniform probability distribution if you do not have any data to support one hypothesis over the other. In addition, it is possible to mathematically prove that our belief will quickly converge to a probability distribution which almost entirely chooses the real environment (see the theorem below). The problem is that the agent's history may be limited, which means that convergence is not rapid enough.

Theorem 4. $\sum_{t=1}^{\infty} |1 - \xi(x_t|x_{1:t-1})| \leq ln(w_x^{-1}) < \infty$ *and* $lim_{t\to\infty}\xi(x_{t:n}|x_t)$ *where* $t \leq n$ *[8, p.24].*

4.2.1 Step 1: Epicurus principal

Epicurus' principle says that we should keep all hypotheses as long as they are consistent with our observed data. This means, as in the example of the experts, that we do not choose to focus on just one expert but continue to listen to all the experts who have been right in their predictions. So the question is, which experts should we listen to from the beginning? Epicurus' principle tells us to listen to all the experts, because all hypothesis can be true before we made a few observations.

This means that we choose the set of all possible environments as our hypothesis class.

Definition 18. *Within the framework of Solomonoff induction, we choose* $M = M_U$ *to be our hypothesis class (ie. the set of all possible programs)[15, p.51].*

4.2.2 Step 2: Occams razor

Occam's razor says that if we have several hypotheses that are consistent with our data, we will choose the simplest hypothesis. This is in contrast to indifference principle which says that we should attribute all hypotheses with the same probability as long as they are consistent with our observations.

You do not have to think long to realize that indifference principle in most cases is absurd because in many cases it is possible to determine the probability of a theory without any observations.

Example:

- **Hypothesis 1:** all emeralds are green.
- **Hypothesis 2:** all emeralds are green until 2020 when they all turn red.

Both hypotheses are consistent with our observations, but instead of giving the hypotheses a uniform probability distribution as the indifference principle suggests, we seem to prefer Occam's razor and give hypothesis 1, a significantly higher probability because it is simpler in the sense that it makes fewer assumptions.

However, there are two problems with Occam's razor. First, it does not define what it means for a hypothesis to be "simple". Second, it seems to go against Epicurus principle because it says that we should throw away the more complicated hypothesis (ie, give them the probability 0), although they are consistent with our observed data. Solomonoff realized that the Kolmogorov complexity was the solution to both problems and come up with the following definition of universal priors.

Definition 19. *The universal prior $P(H_v)^U$ for a hypothesis H_v is $P(H_v)^U := 2^{-K(v)}$ [15, p.50].*

We can see that the definition of universal priors is in line with Occam's razor , because hypotheses that require more number of bits to describe are given a lower probability. In fact, one hypothesis is half as likely if it requires an extra bit in its shortest description.

4.2.3 Justification of Occams razor

Combined, steps 1-2 gives rise to the universal mixture ξ_U.

Definition 20. *The prior probability of a string x is [15, p.52]*

$$\xi_U(x) = \sum_{v \in M_U} P(H_v)^U P(x|H_v)$$

ξ_U is a prior that works for both stochastic and, thus, deterministic environments. But if we, in agreement with the subjectivist stance, suppose that we live in a deterministic world, we can reformulate ξ_U and thereby justify Occam's Razor from the indifference principle. Let us first analyse all programs p with length l with x as output.
$M_l = \{p : len(p) = l\}_{(1)} \Rightarrow |M_l| = 2^l_{(2)} \Rightarrow P(p) = \frac{1}{2^l} = 2^{-l}_{(3)}$
$\Rightarrow P(x) = \sum_{p:U(p)=x*} 2^{-l}_{(4)}$

(1) Is the set of all programs $p \in M_l$ with length l.

(2) Since all programs can be viewed as a binary string, there are 2^l program of length l.

(3) The indifference principle tells us that to be objective we must give a uniform prior probability distribution for each program over M_l. This is done by dividing 1 with the number of programs with length l.

(4) Since we assume that determinism is true, the prior probability of x given all programs with length l, is the sum of the probability of each program which gives x as output.

We may also include programs p where $len(p) \leq l$ by extending p with $l-len(p)$ bits without changing its output. A program p can be extended to have length l in $2^{l-len(p)}$ ways[15, p.53]. We can thus conclude that $2^{l-len(p)}/2^l = 2^{-len(p)}$ is the proportion of programs in M_l that is extended from $len(p)$. This means that shorter program p contributes much more, namely by $2^{-len(p)}$ to the sum above. If we let $l \to \infty$, we can include programs of any length which brings us to the following definition.

Definition 21. $M(x) = \sum_{p:U(p)=x*} 2^{-len(p)}$ *is the universal probability of x.*

Prior probability of a string x can intuitively be seen as the probability that a universal Turing machine U will give x as output when running a program that was randomly generated as follows:

1. We choose a program with a random length l.
2. We choose the program code by the flip of a fair coin l times where the n:th bit x_n in the program is determined by if you get heads or tails the n:th time you flip the coin.

As we just said, it is much more likely for the shortest program to output x because it can be extended in many more ways. The shortest program that gives output x thus dominates the sum above. But the shortest program to output x is $K(x)$ which implies that $M(x)$ is an approximation of $2^{-K(x)}$. So we have used the indifference principle over deterministic environments to justified Occam's razor. [15, p.54-55]

4.2.4 The golden standard

Solomonoff induction was described for the first time in 1964 by Ray Solomonoff, who was a student of Rudolf Carnap. Solomonoff induction is an elegant solution to the Bayesian problem of choosing the priors and hypothesis class.

Besides the problem of priors, there are a host of other problems associated with inductive frameworks such as the confirmation theory problem and raven paradox. Hutter has shown how Solomonoff induction can solve many of these problems.[15]

A big problem with Solomonoff induction is that it is uncomputable because the Kolmogorov complexity of a string is uncomputable [11, p.32]. In other words, Solomonoff induction itself is not a member of M_U. The notion of Solomonoff induction is therfore not compatible with the Church-Turing-Deutsch principle. But the notion of Solomonoff induction is still useful because it can be seen as a golden standard for rational reasoning and all other inductive systems can, in fact, be seen as approximations of Solomonoff induction.[15, p.63-64]

Another problem might be that the Solomonoffs universal prior is a semi-measure and not a real probability measure. This means that the probability distribution over the hypothesis space sum to less or equal to 1 (which makes all

probability measures a subset of all semi-measures). This might make the agent venerable to Dutch book bets. Future research might be needed to establish if that is the case. One might argue that the dependence on the Natural Turing machines also causes problems since the notion of a Natural Turing machines isn't rigorously defined.

There are other ways of dealing with problems of priors within Baynsianism, but Solomonoff induction can, in theory, be seen as the best one. Solomonoff showed that his system of induction was optimal in a universal way because it, by construction, universally dominates all other semi-measures. This means that an agent with Solomonoff induction can assign probabilities to future events with the highest possible level of accuracy, given the data observed, no matter what environment it happens to be in. [21]

We are now in a position where we can define what a reliable inference system is.

Definition 22. *A inference system P is reliable $:\Leftrightarrow$ P is a computable approximation of Solomonoff induction.*

In other words: a reliable inference system is a program $P \in M_U$ such as the agents belief will converge towards $P(\mu) = 1$.

Solomonoff induction has only gained popularity recently as the main ingredient in Marcus Hutters AIXI model of general AI. Although the theory is a synthesis and formalization of the principles of philosophers and scientists such as Epicurus, Empiricus, David Hume, William of Ockham, Pierre-Simon Laplace, Alan Turing and Andrey Kolmogorov, the theory has not yet had any impact on philosophy research. Indeed, if you search on philpapers.org for "Solomonoff" you will only get one hit![1]

4.3 Ways of knowing

It is now possible for us to completely formalize the definition of knowledge according to JTB and RTB.

Definition 23. *Agent A has knowledge that H_v if and only if:*

(i) $H_v = \mu \in M_U$ *(i.e. the agents hypothesis H_v is true if and only if it is a model that corresponds to the real environment that gives rise to the agents observations x)*

(ii) *A have a probability distribution such as* $P(H_v) = 1$

(iii) *A's belief that* $P(H_v) = 1$ *is the result of a inference system P that is a computable approximation of Solomonoff induction.*

We will in the following two sections criticize this knowledge definition from two different standpoints and come up with two new explications of knowledge that are not mutually exclusive but on the contrary complementary.

[1] The search was done on 14-04-2012

4.3.1 Absolute knowledge

Let's pretend for a second that we can stand outside of ourselves as agents in our universe in such a way that we would, in a direct way, be able to see which of all the environments in M_U that our universe truly corresponds to. Let's say that we can simultaneously observe another agent who happens to be a priori preprogrammed with $P(\mu) = 1$ where μ is a complete ToE for our universe (ie. the agent has a full understanding of how every aspect of the environment it inhabits works). There is, therefore, no reason for this agent to update its belief by a reliable inference system, as the agent already is able to predict everything that will happen in the environment with maximum accuracy. It seems absurd to suggest that such an omnipotent agent does not have knowledge. Would we say that God (if he exists) lacks knowledge because he already knows everything there is to know?

Laurence Bonjour notes that the justification is only of importance if we reach the truth, making truth the only thing that has epistemological value.[3, p.8]

So we should theoretically be able to remove (iii) because it does not seem to be a logically necessary condition for knowledge. This is consistent with Crispin Sartwells "epistemic minimalism" which states that an agent S has knowledge of P if and only if S believes in P and P is true [16].We can define this more formally.

Definition 24. *Agent A has absolute knowledge that H_v if and only if:*

(i) $H_v = \mu \in M_U$

(ii) A have a probability distribution such as $P(H_v) = 1$

Sartwell argues that justification is a "criterion" for the testing of knowledge but not a logically necessary "condition" for knowledge itself. What he means by this is that someone may have knowledge without justification, but justification is, however, necessary if we want to test whether someone has the knowledge. This seems a reasonable distinction to make in accordance with our argument above. Sartwell gives the following example: we can test that something is gold by taste when biting on it. But taste is not logically necessary that something should be golden. We can, for example lack taste and it's still gold. The fact that gold has atomic number 79 however is a logical necessary criterion. [16, p.161]

The idea that knowledge should be something testable leads us to our second explication of knowledge.

4.3.2 Generative knowledge

The problem of the epistemic minimalistic definition of knowledge is that we assumed that we had the ability to move outside of ourselves. In reality, we do not have this ability. In fact, an agent will,l by definition, believe that $H_v = mu$ if it has probability distribution $P(H_v) = 1$ regardless of whether this reflects reality or not. Traditionally, philosophers usually have objected to the validity of (iii) as in the section above. It is rare that (i) is ever discussed, and even rarer that anyone questions (i) [18] Our knowledge definition above, based on RTB and JTB, can be rewritten in the following form:
$\{H_v = \mu, B(A, H_v), J(A, H_v)\} \models K(A, H_v)$ where

- $K(A, H_v)$ denote that A has knowledge of H_v
- $B(A, H_v)$ A have a probability distribution such as $P(H_v) = 1$ (i.e. A believes that H_v is true)
- $J(A, H_v)$ denote that A's belief that $P(H_v) = 1$ is the result of a inference system P that is a computable approximation of Solomonoff induction (i.e. the belief is well-founded, or was caused by a reliable process)

The predicate K can be said to be scientifically meaningful only if it is possible to decide (in computability theory sense) whether $K(S, H_v)$ is true or false. Let us now examine whether it is possible to decide the truth value of the claim that an agent A has knowledge of any H_v.

1. $K(A, H_v)$ is the case iff $H_v \wedge B(A, H_v) \wedge J(A, H_v)$ according to the definition of RTB and JTB.
2. We can only know that H_v is the case if we have knowledge of H_v, which we denote $K(Us, H_v)$
3. We can call ourselves A (i.e. assign $A = Us$) and start over at step 1.

We can see that the algorithm ends up in an endless loop (infinite regress) because we do not have a base value (true or false) for H_v. So let us try to find a base value for H_v. The problem with this task is that as soon as we try to make the statement "H_v is true" it is for us indistinguishable from "I believe that H_v is true". "[...]An agent cannot distinguish what she merely believes with justification from what she knows [according to JTB or RTB]" as Mark Kaplan points out [10, p.362]. This is because it is impossible for us to move outside of ourselves and see the truth "directly" (i.e. see directly of what program our universe is running). This makes it impossible to decide whether an agent has knowledge or not as long as truth is a part of the criteria for knowledge.

In fact, the justification of a belief is supposed to be truth conductive. This means that we reach the truth if we justify our belief in a valid manner. If justified belief leads to truth, we get knowledge for free according to JTB and RTB (see figure 5). Everything else other than a justified belief is, thus, redundant.

From a strictly scientific point of view, it is therefore preferable to give up (i) instead of (iii), as we did in our definition of absolute knowledge above. Then it is possible to test a knowledge claim by testing if a process is reliable with another reliable process. If I say "I know that X", your first response is not to ask me if my claim is true, since it is obvious that I think it is, but rather to ask me "how do you know that?". By checking the validity of my reasoning with your internal reasoning and past history it is possible for you to check if my claim is probable.

Let us call this view "generative knowledge", or just knowledge.

Definition 25. *Agent A has generative knowledge that H_v if and only if:*

(i) A have a belief such as $0 < P(H_v) < 1$

(ii) A's belief is the result of a inference system P that is a computable approximation of Solomonoff induction.

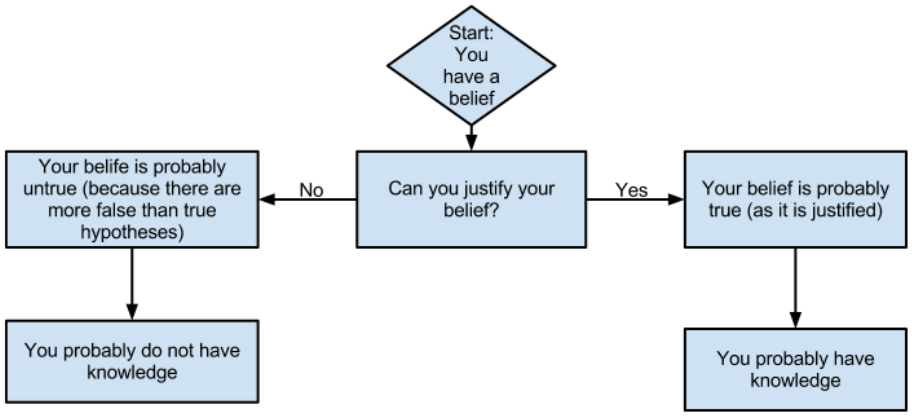

Figure 5: Why everything else than a justified belief is redundant.

It is worth pointing out that generative knowledge is a well calibrated degree of belief rather than knowledge in the usual sense.

But is not also the definition of absolute knowledge above redundant if it is impossible to decide whether an agent has knowledge when one requires that the agent's belief is true?

The answer is no since H_v must by logical necessity be either true or false according to the law of the excluded middle (assuming H_v is empirical and not self-referential like the liar paradox). The definition of absolute knowledge is thus necessary in a logical sense! The point with the argument is not to prove that it is impossible to *have* absolute knowledge or not, in a particular instance but whether it is impossible to *decide* if we have absolute knowledge.

This view of knowledge has the advantage that it is mathematically well defined. It also serves as an intuitive explication, because it reflects the way knowledge is used as a concept in everyday language. One would, for example, not get the idea to claim that Einstein's general relativity or quantum mechanics does not constitute knowledge, because they are not compatible with each other and probably will be replaced with a "theory of everything" sometime in the future. However with our generative definition of knowledge, both quantum mechanics and general relativity are cases of knowledge (although both can not be true simultaneously) because both convictions were created by reliable processes (the scientific method). However, with RTB, JTB or the absolute definition of knowledge only one or none of quantum mechanics and general relativity would constitute knowledge.

4.3.3 The relation between absolute and generative knowledge

Is there a relation between absolute and everyday knowledge?

The agent's generative knowledge will converge towards absolute knowledge if it updates the hypothesis class for an infinite time (see theorem 4).

It is important to note that since belief is a mixture of deterministic environments $v \in M_U$, we must constantly choose what kind of environment we believe that we live in, as if we knew it was the true environment μ with probability 1. For example: Every time we eat there is a small probability that we live in

an environment where our next meal contains a poison with immediate mortality. (i.e. $P(v \in \{p : \text{ no poison in the next meal}\}) < 1$). But we choose every day to act as if we knew that we live in an environment where our next meal is safe (i.e. $P(v \in \{p : \text{ no poison in the next meal}\}) = 1$). It will therefore be instances where the agent has absolute knowledge, despite that the agent beliefs never will reach $P(v \in \{p : \text{ no poison in the next meal}\}) = 1$ in the induction.

But the agent's knowledge is still never absolute in the sense that its actions are risk free since there are uncertainties in every action. It is, therefore, at the same time possible to speak of degrees of knowledge, depending on how long the inference process has lasted. This approach to knowledge is consistent with the scientific method where all truths are considered as provisional and knowledge accumulates over time. It can happen that one may revise or contract their beliefs but in the long run knowledge accumulates.

4.3.4 The value of generative and absolute knowledge

The idea that knowledge must be something valuable dates back to Plato's famous argument in the dialogue Meno.

But RTB and JTB is no more valuable than true belief according to an argument called the swamping argument. Erik Olsson presents the swamping argument schematically in the following way:[13, p.94]

(S1) Knowledge equals reliably produced true belief (simple reliabilism).

(S2) If a given belief is true, its value will not be raised by the fact that it was reliably produced.

(S1) Hence: knowledge is no more valuable than unreliably produced true belief.

The value of reliability is therefore swamped by the value of truth. The swamping argument states in other words, if we use our knowledge explications, that generative knowledge is no more useful than absolute knowledge. A key to the solution is the fact that RTB knowledge has, according to David Armstrong, an advantage over the merely true belief because a reliable process can be employed more than once and produce the desired result[1, p.173].

A Chinese proverb goes "Give a man a fish and you feed him for a day. Teach a man to fish and you feed him for a lifetime".

This proverb captures the essence why a reliable true belief is more valuable then just true belief. An agent might have the true belief that he will somehow get a fish at time t (someone gives it to him), but that instance of knowledge was probably just pure luck. If the agent instead has a reliable process of how to get fishes by knowing how to fish, then this method can be employed every time the agent is hungry.

The swamping argument does not seem to be valid for that reason. Olsson call this solution to the swamping problem the "conditional probability solution" because the probability of future true belief is more conditional on knowledge according to RTB as opposed to conditional on mere true belief [13].

But this is only true for partial hypothesis. The swamping argument will be valid again if the hypothesis happens to be a true ToE q. This is because an agent that has a belief in q will be able to make all correct predictions, as we noted in section 4.3.1.

This counter argument is valid given that the hypothesis really is q, but it is at the same time problematic for the following two reasons:

1. It is extremely unlikely that an agent will choose q by random because the universal probability for q is $2^{-K(q)}$.

2. It will simply not be possible for the agent to compute q because it would seemingly require as much memory as the entire universe contains to compute the next position of every particle in the universe. This is simply not possible since the agent is a part of the universe it inhabits (as long as the agent does not have access to computational resources beyond its universe).

It is therefore safe to say that generative knowledge is more valuable than absolute knowledge.

Olsson also puts forward a list of empirical regularities that must hold for the conditional probability solution to be true.[14, p.879-880]

- "Non-uniqueness: once you encounter a problem of a certain type, you are likely to face other problems of the same type in the future."

- "Cross-temporal access: a method that was used once is often available when similar problems arise in the future."

- "Learning: a method that was unproblematically employed once will tend to be employed again on similar problems in the future."

- "Generality: a method that is (un)reliable in one situation is likely to be (un)reliable in other similar situations in the future."

The two first regularities are related to the world while the last two are related to the agent. It is important that the two last regularities hold for Solomonoff induction since we have defined a reliability as the computational approximation of Solomonoff induction. Otherwise, the argument why generative knowledge is more valuable than absolute knowledge will not be valid.

It is obvious that an agent implemented with Solomonoff induction will learn since it is a learning algorithm. The question of generality of Solomonoff induction is related to the "no free lunch theorem" in artificial intelligence. The free lunch theorem basically states that all inference systems P preform equally poorly over all environments in the environment class M_U. Hutter has shown that the "no free lunch theorem" is false and does therefore not apply for Solomonoff induction.[15, p.10].

5 Summary

In this section we will briefly summarize our digital ontology and epistemology.

5.1 Ontology

We can summarize the digital ontology in the table below.

The World (U, X, M_U)	**The Agent** $(P, [0,1], X, M)$
$U : M_U \to X^\infty \Leftrightarrow$ $U : \{0,1\}^* \to \{0,1\}^\infty$	$P : X^\infty \to M \to [0,1] \Leftrightarrow$ $P : \{0,1\}^\infty \to \{0,1\}^* \to [0,1]$
The world contains a program $u \in M_U$ that is generating a universe x from the atoms in X when running the Turing machine U.	The agent contains a sensation history $x \in X$ that is generating a belief $P(v, x) \in [0,1]$ when it updates with the function P.
This is analogous to engineering because we have an environment v as input and generate a universe history x as output.	This is analogous to reversed engineering because we have a history x as input and give our belief in what environment v caused the history as output.

Table 3: Overview of the digital ontology.

One possible criticism of the digital philosophy that has been presented in this paper is that it is far too idealized and does not comply with a more complex philosophical reality. This is not true because every possible (computable) formal ontology and instantiated laws of physics is a part of (U, X, M_U). Even the agent is a subsystem of the world, where $P \in M$ according to the Church-Turing-Deutsch principle, because the universe would no longer be computable if it contained an uncomputable agent. All possible (computable) epistemologies and all of the agent's instantiated mental concepts are in the same way emergent from $(P, [0,1], X, M)$ for the same reason.

If the Church-Turing-Deutsch principle would prove to be false then the entire model collapses. This possibility is, of course, not to be ruled out.

It is worth noting that if we encode the $[0,1]$ to binary numbers then everything in both the world including the agent, can be said to belong the set $\{0,1\}^\infty$. Hence the name digital philosophy!

5.2 Epistemology

The definition of absolute knowledge, RTB and JTB are practically useless because it is not possible to decide whether anyone has knowledge or not and it has lower expected value. But it is still useful from a epistemic point of view because it assures that there is an objective truth our generative knowledge will converge to in the limit.

When we say that we have knowledge of A, we want not only A to be a reliable belief (generative knowledge), but we also want A to correspond to reality (absolute knowledge). This fact makes generative and absolute knowledge mutually inclusive.

Absolute knowledge	**Generative knowledge**
Agent A has absolute knowledge that H_v if and only if: (i) $H_v = \mu \in M_U$ (ii) A have a probability distribution such as $P(H_v) = 1$	Agent A has knowledge that H_v if and only if: (i) A have a belief such as $0 < P(H_v) < 1$ (ii) A's belief is the result of a inference system P that is a computable approximation of Solomonoff induction.
It is impossible to test if an agent has knowledge.	It is possible to test if an agent has knowledge by checking what inference system it used to draw its conclusions.
Is useful only in *philosophy* as a reassurance that there is an objective truth but it has a low expected value when randomly selected.	Is useful in *science* as a the generator of true belief in the limit and has therefore high expected value when randomly selected.

Table 4: Overview of the digital epistemology.

But ultimately, we can not perform better than to try to justify our belief and hope that it results from a reliable process that is truth conductive (it is always possible that we have some unconscious bias). It is therefore important to separate the explication of absolute and generative knowledge since absolute knowledge is ultimately consciously unreachable (i.e. undecidable), even if we very well might have it at any given point in time (and expect to have it more often the more data we collect).

As a scientist, it is not fruitful to ask for truth, but rather to ask for a reliable reasoning from the evidence. Generative knowledge is, therefore, the only useful explication from a purely scientific point of view. The definition of generative knowledge should therefore be stipulated in the spirit of Carnap's idea of pragmatic explications. [2].

RTB and JTB, on the other hand, capture neither the epistemic necessary properties of absolute knowledge (as we showed in section 4.3.1) or the useful properties of generative knowledge (as we showed in section 4.3.2).

References

[1] D.M. Armstrong. *Belief, Truth and Knowledge*. Cambridge University Press, 1973.

[2] M. Beaney. Analysis. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Summer 2012 edition, 2012.

[3] L. BonJour. *The Structure of Empirical Knowledge*. Harvard University Press, 1985.

[4] G.J. Chaitin. *Algorithmic Information Theory*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2004.

[5] B. Jack Copeland. The church-turing thesis. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Fall 2008 edition, 2008.

[6] D. Deutsch. *The Fabric of Reality*. Penguin Books Limited, 1998.

[7] G.W. Flake. *The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaptation*. A Bradford Book. Mit Press, 2000.

[8] M. Hutter. A complete theory of everything (will be subjective). *CoRR*, abs/0912.5434, 2009.

[9] A. Hájek. Interpretations of probability. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Summer 2012 edition, 2012.

[10] M. Kaplan. It's not what you know that counts. *The Journal of Philosophy*, 82(7):350–363, Jul. 1985.

[11] S. Legg. *Machine Super Intelligence*. PhD thesis, 2008.

[12] S. Lloyd. *Programming the Universe: A Quantum Computer Scientist Takes On the Cosmos*. Knopf, 2006.

[13] E.J. Olsson. In defense of the conditional probability solution to the swamping problem. *Grazer Philosophische Studien*, 79(1):93 – 114, 2009.

[14] E.J Olsson. The value of knowledge. *Philosophy Compass*, 6(12):874–883, 2011.

[15] S. Rathmanner and M. Hutter. A philosophical treatise of universal induction. *CoRR*, abs/1105.5721, 2011.

[16] C. Sartwell. Knowledge is merely true belief. *American Philosophical Quarterly*, 28:157–165, Apr. 1991.

[17] J. Schmidhuber. Algorithmic theories of everything. `http://arxiv.org/abs/quant-ph/0011122`, 2000.

[18] M. Steup. The analysis of knowledge. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Summer 2012 edition, 2012.

[19] G. 't Hooft. Quantum gravity as a dissipative deterministic system. `http://arxiv.org/abs/gr-qc/9903084`, 1999.

[20] M. Tegmark. The mathematical universe. *Foundations of Physics*, 38, 2008.

[21] P.M.B. Vitanyi, M. Hutter, and S. Legg. Algorithmic probability. *Scholarpedia*, 2(8):2572, 2007.

[22] R. Von Mises. *Probability, Statistics, and Truth*. Dover books explaining science. Dover Publications, 1981.

[23] S. Wolfram. *A new kind of science*. General science. Wolfram Media, 2002.

[24] S. Zuse. Calculating space. `ftp://ftp.idsia.ch/pub/juergen/zuserechnenderraum.pdf`, 1969.