A Formal Approach to the Problem of Logical Non-Omniscience

Scott Garrabrant Tsvi Benson-Tilsen Andrew Critch
Nate Soares Jessica Taylor
Machine Intelligence Research Institute
Berkeley, CA
{scott,tsvi,critch,nate,jessica}@intelligence.org

We present the *logical induction criterion* for computable algorithms that assign probabilities to every logical statement in a given formal language, and refine those probabilities over time. The criterion is motivated by a series of stock trading analogies. Roughly speaking, each logical sentence ϕ is associated with a stock that is worth \$1 per share if ϕ is true and nothing otherwise, and we interpret the belief-state of a logically uncertain reasoner as a set of market prices, where $\mathbb{P}_n(\phi) = 50\%$ means that on day n, shares of ϕ may be bought or sold from the reasoner for $50¢$. A market is then called a *logical inductor* if (very roughly) there is no polynomial-time computable trading strategy with finite risk tolerance that earns unbounded profits in that market over time. We then describe how this single criterion implies a number of desirable properties of bounded reasoners; for example, logical inductors outpace their underlying deductive process, perform universal empirical induction given enough time to think, and place strong trust in their own reasoning process.

1 Introduction

Every student of mathematics has experienced uncertainty about conjectures for which there is "quite a bit of evidence", such as the Riemann hypothesis or the twin prime conjecture. Indeed, when Zhang [51] proved a bound on the gap between primes, we were tempted to increase our credence in the twin prime conjecture. But how much evidence does this bound provide for the twin prime conjecture? Can we quantify the degree to which it should increase our confidence?

The natural impulse is to appeal to probability theory in general and Bayes' theorem in particular. Bayes' theorem gives rules for how to use observations to update empirical uncertainty about unknown events in the physical world.

However, probability theory lacks the tools to manage logical non-omniscience: probability-theoretic reasoners cannot possess uncertainty about logical facts so long as their beliefs respect basic logical constraints. For example, let ϕ stand for the claim that the 87,653rd digit of π is a 7. If this claim is true, then $(1+1=2) \Rightarrow \phi$. But the laws of probability theory say that if $A \Rightarrow B$ then $\Pr(A) \leq \Pr(B)$. Thus, a perfect Bayesian must be at least as sure of ϕ as they are that $1+1=2$! Recognition of this problem dates at least back to [23].

Many have proposed methods for relaxing the criterion $\Pr(A) \leq \Pr(B)$ until such a time as the implication has been proven (see, e.g., the work of [26, 7]). But this leaves open the question of how probabilities should be assigned before the implication is proven, and this brings us back to the search for a principled method for managing uncertainty about logical facts when relationships between them are suspected but unproven.

In this paper we describe what we call the *logical induction criterion* for reasoning under logical uncertainty. Our solution works, more or less, by treating a reasoner's beliefs as prices in a market

J. Lang (Ed.): TARK 2017
EPTCS 251, 2017, pp. 221–235, doi:10.4204/EPTCS.251.16

that fluctuate over time, and requiring that those prices not be exploitable indefinitely by any sequence of trades constructed by an efficient (polynomial-time) algorithm. The logical induction criterion can be seen as a weakening of the "no Dutch book" criteria that Ramsey [42], de Finetti [14], Teller [48], and Lewis [36] used to support standard probability theory, which is analogous to the "no Dutch book" criteria that von Neumann and Morgenstern [40] and Joyce [33] used to support expected utility theory. Because of the analogy, and the variety of desirable properties that follow immediately from this one criterion, we believe that the logical induction criterion captures a portion of what it means to do good reasoning about logical facts in the face of deductive limitations.

Section 2 lists desiderata for reasoning under logical uncertainty.

Section 3 lists further related work.

Section 4 presents an overview of the logical induction framework.

Section 5 discusses a collection of properties satisfied by logical inductors.

Section 6 gives concluding remarks.

Note on abridgement: Due to space considerations, this paper does not include proofs of claims, and describes some results only at a high level. The formal details of our definitions and theorems, additional properties of logical inductors, proofs of properties, a construction of a logical inductor, and further discussion can be found in [18].

2 Desiderata for Reasoning under Logical Uncertainty

For historical context, and to further reify the problem, we now review a number of desiderata that have been proposed in the literature as desirable features of "good reasoning" in the face of logical uncertainty.

Desideratum 1 (Computable Approximability)**.** *The method for assigning probabilities to logical claims (and refining them over time) should be computable.*

Desideratum 2 (Coherence in the Limit)**.** *The belief state that the reasoner is approximating better and better over time should be logically consistent.*

(Discussed in Section 5.2.)

Desideratum 3 (Approximate Coherence)**.** *The belief states of the reasoner over time should be approximately logically consistent.*

(Discussed in Section 5.3.)

Desideratum 3 dates back to at least Good [23], who proposes a weakening of the condition of coherence that could apply to the belief states of limited reasoners. Hacking [26] proposes an alternative weakening, as do Garrabrant et al. [19].

Desideratum 4 (Learning of Statistical Patterns)**.** *In lieu of knowledge that bears on a logical fact, a good reasoner should assign probabilities to that fact in accordance with the rate at which similar claims are true.*

For example, a good reasoner should assign probability $\approx 10\%$ to the claim "the nth digit of π is a 7" for large n (assuming there is no efficient way for a reasoner to guess the digits of π for large n); see [44].

Desideratum 5 (Calibration)**.** *Good reasoners should be well-calibrated. That is, among events that a reasoner says should occur with probability p, they should in fact occur about p proportion of the time.*

Desideratum 6 (Non-Dogmatism)**.** *A good reasoner should not have extreme beliefs about mathematical facts, unless those beliefs have a basis in proof.*

(Discussed in Section 5.2.)

In the domain of logical uncertainty, Desideratum 6 can be traced back to Carnap [6, Sec. 53], and has been demanded by many, including Gaifman[16] and Hutter [31].

Desideratum 7 (Uniform Non-Dogmatism)**.** *A good reasoner should assign a non-zero probability to any computably enumerable consistent theory (viewed as a limit of finite conjunctions).*

(Discussed in Section 5.2.)

The first formal statement of Desideratum 7 that we know of is given by Demski [9], though it is implicitly assumed whenever asking for a set of beliefs that can reason accurately about arbitrary arithmetical claims (as is done by, e.g., Savage [44] and Hacking [26]).

Desideratum 8 (Universal Inductivity)**.** *Given enough time to think, the beliefs of a good reasoner should dominate any (lower semicomputable) semimeasure.*

(Discussed in Section 5.2.)

Desideratum 9 (Approximate Bayesianism)**.** *The reasoner's beliefs should admit of some notion of conditional probabilities, which approximately satisfy both Bayes' theorem and the other desiderata listed here.*

Desideratum 10 (Self-knowledge)**.** *If a good reasoner knows something, she should also know that she knows it.* (Discussed in Section 5.4.)

Proposed by Hintikka [30], Desideratum 10 is popular among epistemic logicians. This desideratum has been formalized in many different ways; see [8, 5] for a sample.

Desideratum 11 (Self-Trust)**.** *A good reasoner thinking about a hard problem should expect that, in the future, her beliefs about the problem will be more accurate than her current beliefs.* (Discussed in Section 5.5.)

Desideratum 12 (Approximate Inexploitability)**.** *It should not be possible to run a Dutch book against a good reasoner in practice.* (See Section 4 for our proposal.)

As noted by Eells [10], the Dutch book constraints used by von Neumann and Morgenstern [40] and de Finetti [14] are implausibly strong: all it takes to run a Dutch book according to de Finetti's formulation is for the bookie to know a logical fact that the reasoner does not know. Thus, to avoid being Dutch booked by de Finetti's formulation, a reasoner must be logically omniscient.

Hacking [26] and Eells [10] call for weakenings of the Dutch book constraints, in the hopes that reasoners that are approximately inexploitable would do good approximate reasoning. This idea is the cornerstone of our framework—we consider reasoners that cannot be exploited by betting strategies that can be constructed by a polynomial-time machine.

Logical inductors satisfy desiderata 1 through 12. In fact, logical inductors are designed to meet only Desideratum 1 (computable approximability) and Desideratum 12 (approximate inexploitability), from which 2-11 all follow (see [18]).

3 Additional Related Work

The study of logical uncertainty is an old topic. It can be traced all the way back to Bernoulli, who laid the foundations of statistics, and later Boole [4], who was interested in the unification of logic with probability from the start. Refer to [27] for a historical account. Our algorithm assigns probabilities to sentences of logic directly; this thread can be traced back through Łoś [38] and later Gaifman [15], who developed the notion of coherence that we use in this paper.

When it comes to the problem of developing formal tools for manipulating uncertainty, our methods are heavily inspired by Bayesian probability theory, and so can be traced back to Pascal, who was followed by Bayes, Laplace, Kolmogorov [34], Savage [43], Carnap [6], Jaynes [32], and many others. Polya [41] was among the first in the literature to explicitly study the way that mathematicians engage in plausible reasoning, which is tightly related to the object of our study.

In addition to Good [23], Savage [44], and Hacking [26], the flaw in Bayesian probability theory was also highlighted by Glymour [21], and dubbed the "problem of old evidence" by Garber [17] in response to Glymor's criticism. Eells [10] gave a lucid discussion of the problem, revealed flaws in Garber's arguments and in Hacking's solution, and named a number of other desiderata which our algorithm manages to satisfy; see [53] and [47]. Adams [2] uses logical deduction to reason about an unknown probability distribution that satisfies certain logical axioms. Our approach works in precisely the opposite direction: we use probabilistic methods to create an approximate distribution where logical facts are the subject.

Some work in epistemic logic has been directed at modeling the dynamics of belief updating in non-omniscient agents; see for example [35, 50, 3]. Our approach differs in that we use first-order logic, and therefore use the recursion theorem to make reflective statements instead of using explicit knowledge or belief operators; the potential paradoxes of self-reference are circumvented by allowing beliefs to be probabilistic. The mechanism used by our logical inductor to update its beliefs is not very transparent, leaving open the possibility of a more principled understanding of the local mechanics of updating probabilities on logical or inductive inferences.

Straddling the boundary between philosophy and computer science, Aaronson [1] has made a compelling case that computational complexity must play a role in answering questions about logical uncertainty. Fagin and Halpern [12] also straddled this boundary with early discussions of algorithms that manage uncertainty in the face of resource limitations. (See also their discussions of uncertainty and knowledge. [13, 28])

4 The Logical Induction Criterion

We propose a partial solution to the problem of logical non-omniscience, which we call *logical induction*. Roughly speaking, a *logical inductor* is a computable reasoning process that is not exploitable by any polynomial-time computable strategy for making trades against it, using its probabilities as the prices of shares. In this section we give a high-level overview of the criterion and the main result (details are in [18]), before giving precise statements in Section 5 of some of the properties satisfied by logical inductors.

Very roughly, our setup works as follows. We consider reasoners that assign probabilities to sentences $\mathscr{S}$ written in some formal language $\mathscr{L}$.

Definition 4.0.1 (Pricing)**.** *A **pricing** is a computable rational function $\mathbb{P} : \mathscr{S} \to \mathbb{Q} \cap [0,1]$.*

Here $\mathbb{P}(\phi)$ is interpreted as the probability of ϕ. We can visualize a pricing as a list of (ϕ, p) pairs, where the ϕ are unique sentences and the p are rational-number probabilities, and $\mathbb{P}(\phi)$ is defined to be p if (ϕ, p) occurs in the list, and 0 otherwise. (In this way we can represent belief states of reasoners that can be written down explicitly in a finite amount of space.) The output of a reasoner is then nothing but a sequence of pricings:

Definition 4.0.2 (Market)**.** *A **market** $\overline{\mathbb{P}} = (\mathbb{P}_1, \mathbb{P}_2, \ldots)$ is a computable sequence of pricings $\mathbb{P}_i : \mathscr{S} \to \mathbb{Q} \cap [0,1]$.*

The pricings $(\mathbb{P}_1, \mathbb{P}_2, \ldots)$ represent the belief states of a reasoner progressively refining their opinions about the logical statements in $\mathscr{S}$. In the background, there is some process producing progressively larger sets of trusted statements:

Definition 4.0.3 (Deductive Process)**.** *A **deductive process** $\overline{D} : \mathbb{N}^+ \to \mathrm{Fin}(\mathscr{S})$ is a computable nested sequence $D_1 \subseteq D_2 \subseteq D_3 \ldots$ of finite sets of sentences.*

The deductive process $\overline{D}$ can be thought of as a theorem prover for some trusted logical theory Γ in the language $\mathscr{L}$. Indeed, we will henceforth assume that $\Gamma = \bigcup_n D_n$. Thus the goal of our reasoner $\overline{\mathbb{P}}$ is to anticipate which statements will be proven or disproven by Γ, well before the rote proof-search $\overline{D}$ decides those statements.

As in classical Dutch book arguments for probability theory, in addition to seeing $\mathbb{P}(\phi) = p$ as an assignment of subjective credence to ϕ, we also view $\mathbb{P}(\phi)$ as a stance with respect to which bets are desirable or not. That is, we interpret $\mathbb{P}(\phi) = p$ to mean that the price of a ϕ-share according to $\mathbb{P}$ is p, where (roughly speaking) a ϕ-share is worth \$1 if ϕ is true. This allows us to set up Dutch book arguments against a reasoner using computable bookies:

Definition 4.0.4 (Trader)**.** *A **trader** is a sequence $(T_1, T_2, \ldots)$ where each T_n is a trading strategy for day n.*

Without belaboring the details, a trading strategy for day n is a strategy for responding to the day's market prices $\mathbb{P}_n$ with buy orders and sell orders for shares in sentences from $\mathscr{S}$. (Formally, it is a continuous function from pricings to linear combinations of sentences, expressed in some computable language.) Over time, a trader accumulates cash and stock holdings from the trades it makes against $\overline{\mathbb{P}}$.

The logical induction criterion then demands of market prices $\overline{\mathbb{P}}$ that no efficiently computable trader can reliably make money by trading against the market prices $(\mathbb{P}_1, \mathbb{P}_2, \ldots)$:

Definition 4.0.5 (The Logical Induction Criterion)**.** *A market $\overline{\mathbb{P}}$ is said to satisfy the **logical induction criterion** relative to a deductive process $\overline{D}$ if there is no efficiently computable trader that exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$. A market $\overline{\mathbb{P}}$ meeting this criterion is called a **logical inductor over** $\overline{D}$.*

Again glossing over details, a trader is said to exploit $\overline{\mathbb{P}}$ relative to $\overline{D}$ if the possible values of the trader's holdings from trading against $\overline{\mathbb{P}}$ are unboundedly high over time, without being unboundedly low, where holdings are evaluated by what truth assignments to $\mathscr{S}$ are propositionally consistent with D_n at time n. Here, "efficiently computable" (abbreviated e.c.) can be taken to mean computable in time polynomial in n, but this is not crucial to the definition. Given the assumption that $\Gamma = \bigcup_n D_n$, we also say that $\overline{\mathbb{P}}$ is a logical inductor over Γ.

Our key theorem is that this criterion, while gratifyingly strong, is also feasible:

Theorem 4.0.6. *For any deductive process $\overline{D}$, there exists a computable belief sequence $\overline{\mathbb{P}}$ satisfying the logical induction criterion relative to $\overline{D}$.*

5 Properties of Logical Inductors

Here is an intuitive argument that logical inductors perform good reasoning under logical uncertainty:

> Consider any polynomial-time method for efficiently identifying patterns in logic. If the market prices don't learn to reflect that pattern, a clever trader can use that pattern to exploit the market. Thus, a logical inductor must learn to identify those patterns.

This section will substantiate this argument by stating a number of properties satisfied by logical inductors, corresponding to some of the desiderata discussed in Section 2. Proofs of Theorem 4.0.6 and the theorems in this section can be found in [18].

5.1 Notation

Throughout, we assume that $\overline{\mathbb{P}}$ is a logical inductor over the theory Γ. We also assume that Γ represents computations in the technical sense, i.e. we can write terms in $\mathscr{L}$ that stand for computations, and Γ proves that those terms evaluate to their correct value (and no other value).

We will enclose sentences in quotation marks when they are used as syntactic objects. An underlined symbol should be replaced by the expression it stands for. For example, $\underline{f(n)}$ stands for a program that computes the function f given input n, whereas $\underline{f(n)}$ stands for the numeral $f(n)$ evaluates to.

We use an overline to denote sequences of sentences, probabilities, and other objects, as in $\overline{\mathbb{P}}$ and $\overline{D}$; for example, $\overline{\phi}$ is the sequence of sentences $(\phi_1, \phi_2, \dots)$. A sequence $\overline{x}$ is efficiently computable (e.c.) if and only if there exists a computable function $n \mapsto x_n$ with runtime polynomial in n. Given any sequences $\overline{x}$ and $\overline{y}$, we write

$$x_n \eqsim_n y_n \quad \text{for} \quad \lim_{n\to\infty} x_n - y_n = 0, \text{ and}$$
$$x_n \gtrsim_n y_n \quad \text{for} \quad \liminf_{n\to\infty} x_n - y_n \geq 0.$$

5.2 Properties of the limit

Firstly, the market prices of a logical inductor converge:

Theorem 5.2.1 (Convergence)**.** *The limit* $\mathbb{P}_\infty : \mathscr{S} \to [0,1]$ *defined by*

$$\mathbb{P}_\infty(\phi) := \lim_{n\to\infty} \mathbb{P}_n(\phi)$$

exists for all ϕ*.*

Proof sketch.

> Roughly speaking, if $\overline{\mathbb{P}}$ never makes up its mind about ϕ, then it can be exploited by a trader arbitraging shares of ϕ across different days. That is, suppose by way of contradiction that $\mathbb{P}_n(\phi)$ never settles down, but rather oscillates by a substantial amount infinitely often. Then there is a trader that repeatedly buys a share in ϕ when the price is low, and sells it back when the price is high. This trader accumulates unbounded wealth, thereby exploiting $\overline{\mathbb{P}}$, which contradicts that $\overline{\mathbb{P}}$ is a logical inductor; therefore the limit $\mathbb{P}_\infty(\phi)$ must in fact exist.

This sketch showcases the main intuition for the convergence of $\overline{\mathbb{P}}$, but elides a number of crucial details; see [18].

Next, the limiting beliefs of a logical inductor represent a coherent probability distribution:

Theorem 5.2.2 (Limit Coherence)**.** $\mathbb{P}_\infty$ *is coherent, i.e., it gives rise to an internally consistent probability measure* Pr *on the set of all consistent completions* $\Gamma' : \mathscr{S} \to \mathbb{B}$ *of* Γ*, defined by the formula*

$$\Pr(\Gamma'(\phi) = 1) := \mathbb{P}_\infty(\phi).$$

First formalized by Gaifman [15], coherence says that beliefs should satisfy probabilistic versions of logical consistency; for example, the reasoner should assign $\Pr(\phi) \leq \Pr(\psi)$ if $\phi \Rightarrow \psi$, etc. This theorem is proven using methods analogous to standard Dutch book arguments for coherent beliefs, translated into the language of traders.

Convergence and coherence together justify that a logical inductor $\overline{\mathbb{P}}$ approximates a belief state that is consistent with the background theory Γ. What else is there to say about the limiting beliefs $\mathbb{P}_\infty$ of a logical inductor?

For starters, $\overline{\mathbb{P}}$ learns not to assign extreme probabilities to sentences that are independent from Γ:

Theorem 5.2.3 (Non-Dogmatism)**.** *If $\Gamma \nvdash \phi$ then $\mathbb{P}_\infty(\phi) < 1$, and if $\Gamma \nvdash \neg\phi$ then $\mathbb{P}_\infty(\phi) > 0$.*

Non-dogmatism can be viewed as an inductive property: non-dogmatic beliefs can be easily conditioned on events (sentences) that haven't already been observed (proved or disproved), producing a coherent conditional belief state, whereas conditioning dogmatic beliefs can cause problems.

We can push the idea of inductive reasoning much further, following the work of Solomonoff [45, 46], Zvonik and Levin [52] and Li and Vitányi [37] on empirical sequence prediction. They describe an inductive process (known as a universal semimeasure) that predicts as well or better than any computable predictor, modulo a constant amount of error. Although universal semimeasures are uncomputable, we can ask logically uncertain reasoners to copy those successes given enough time to think:

Theorem 5.2.4 (Domination of the Universal Semimeasure)**.** *Let $(b_1, b_2, \ldots)$ be a sequence of zero-arity predicate symbols in $\mathscr{L}$ not mentioned in Γ, and let $\sigma_{\leq n} = (\sigma_1, \ldots, \sigma_n)$ be any finite bitstring. Define*

$$\mathbb{P}_\infty(\sigma_{\leq n}) := \mathbb{P}_\infty(\text{“}(b_1 \leftrightarrow \underline{\sigma_1} = 1) \wedge \ldots \wedge (b_n \leftrightarrow \underline{\sigma_n} = 1)\text{”}),$$

such that, for example, $\mathbb{P}_\infty(01101) = \mathbb{P}_\infty(\text{“}\neg b_1 \wedge b_2 \wedge b_3 \wedge \neg b_4 \wedge b_5\text{”})$. Let M be a universal continuous semimeasure. Then there is some positive constant C such that for any finite bitstring $\sigma_{\leq n}$,

$$\mathbb{P}_\infty(\sigma_{\leq n}) \geq C \cdot M(\sigma_{\leq n}).$$

In other words, logical inductors are a computable approximation to a normalized probability distribution that dominates any lower semicomputable semimeasure. In fact, this dominance is strict: $\mathbb{P}_\infty$ will e.g., assign positive probability to sequences that encode completions of Peano arithmetic, which the universal semimeasure does not do.[1]

5.3 Outpacing deduction

It is not too difficult to define a reasoner that assigns probability 1 to all (and only) the provable sentences, in the limit: simply assign probability 0 to all sentences, and then enumerate all logical proofs, and assign probability 1 to the proven sentences. The real trick is to recognize patterns in a timely manner, well before the sentences can be proven by slow deduction.

Theorem 5.3.1 (Provability Induction)**.** *Let $\overline{\phi}$ be an e.c. sequence of theorems. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

Furthermore, let $\overline{\psi}$ be an e.c. sequence of disprovable sentences. Then

$$\mathbb{P}_n(\psi_n) \eqsim_n 0.$$

[1] This does not contradict the universality of M, as $\mathbb{P}_\infty$ is higher in the arithmetical hierarchy than M.

Proof sketch.

> Suppose not. Then there is a trader that buys a share in ϕ_n whenever the price is too far below \$1, and then waits for ϕ_n to appear in the deductive process $\overline{D}$, repeating this process indefinitely. This trader would exploit $\overline{\mathbb{P}}$, a contradiction.

In other words, $\overline{\mathbb{P}}$ will learn to start believing ϕ_n by day n at the latest, despite the fact that ϕ_n won't be deductively confirmed until day $f(n)$, which is potentially much later. In colloquial terms, if $\overline{\phi}$ is a sequence of facts that can be generated efficiently, then $\overline{\mathbb{P}}$ inductively learns the pattern, and its belief in $\overline{\phi}$ becomes accurate faster than $\overline{D}$ can computationally verify the individual sentences.

> **Analogy: Ramanujan and Hardy.** Imagine that the statements $\overline{\phi}$ are being output by an algorithm that uses heuristics to generate mathematical facts without proofs, playing a role similar to the famously brilliant, often-unrigorous mathematician Srinivasa Ramanujan. Then $\overline{\mathbb{P}}$ plays the historical role of the beliefs of the rigorous G.H. Hardy who tries to verify those results according to a slow deductive process ($\overline{D}$). After Hardy ($\overline{\mathbb{P}}$) verifies enough of Ramanujan's claims ($\phi_{\leq n}$), he begins to trust Ramanujan, even if the proofs of Ramanujan's later conjectures are incredibly long, putting them ever-further beyond Hardy's current abilities to rigorously verify them. In this story, Hardy's inductive reasoning (and Ramanujan's also) outpaces his deductive reasoning.

To further emphasize the meaning of Theorem 5.3.1 (Provability Induction), consider the famous halting problem of Turning [49]. Turing proved that there is no general algorithm for determining whether or not an arbitrary computation halts. Let's examine what happens when we confront logical inductors with the halting problem.

Theorem 5.3.2 (Learning of Halting Patterns)**.** *Let $\overline{m}$ be an e.c. sequence of Turing machines, and $\overline{x}$ be an e.c. sequence of bitstrings, such that m_n halts on input x_n for all n. Then*

$$\mathbb{P}_n(\text{“}\underline{m_n} \textit{ halts on input } \underline{x_n}\text{”}) \eqsim_n 1.$$

Of course, this is not so hard on its own—a function that assigns probability 1 to everything also satisfies this property. The real trick is separating the halting machines from the non-halting ones.

By undecidability, there are Turing machines q that fail to halt on input y, but such that Γ is not strong enough to prove this fact. In this case, $\mathbb{P}_\infty$'s probability of q halting on input y is positive, by Theorem 5.2.3 (Non-Dogmatism). Nevertheless, $\overline{\mathbb{P}}$ still learns to stop expecting that those machines will halt after any reasonable amount of time:

Theorem 5.3.3 (Learning not to Anticipate Halting)**.** *Let $\overline{q}$ be an e.c. sequence of Turing machines, and let $\overline{y}$ be an e.c. sequence of bitstrings, such that q_n does not halt on input y_n for any n. Let f be any computable function. Then*

$$\mathbb{P}_n(\text{“}\underline{q_n} \textit{ halts on input } \underline{y_n} \textit{ within } \underline{f}(\underline{n}) \textit{ steps}\text{”}) \eqsim_n 0.$$

These theorems can be interpreted as justifying the intuitions that many computer scientists have long held towards the halting problem: It is impossible to tell whether or not a Turing machine halts in full generality, but for large classes of well-behaved computer programs (such as e.c. sequences of halting programs and provably non-halting programs) it's quite possible to develop reasonable and accurate beliefs. The boundary between machines that compute fast-growing functions and machines that never halt is difficult to distinguish, but even in those cases, it's easy to learn to stop expecting those machines to halt within any reasonable amount of time.

As a consequence of of Theorem 5.3.3 (Learning not to Anticipate Halting), a logical inductor will trust their (computable) underlying deductive process $\overline{D}$ to remain consistent for arbitrarily long specified periods of time, if in fact $\overline{D}$ is consistent. In other words, a logical inductor over the theory Γ will learn trust in the finitary consistency of Γ.

One possible objection here is that the crux of the halting problem (and of the Γ-trust problem) is not about making good predictions, it is about handling diagonalization and paradoxes of self-reference. So let us turn to the topic of $\overline{\mathbb{P}}$'s beliefs about $\overline{\mathbb{P}}$ itself.

5.4 Self-knowledge

Because we're assuming Γ can represent computable functions, we can write sentences describing the beliefs of $\overline{\mathbb{P}}$ at different times. What happens when we ask $\overline{\mathbb{P}}$ about sentences that refer to itself?

Theorem 5.4.1 (Self-knowledge)**.** *Let $\overline{\phi}$ be an e.c. sequence of sentences, let $\overline{a}$, $\overline{b}$ be e.c. sequences of probabilities. Then, for any e.c. sequence of positive rationals $\overline{\delta} \to 0$, there exists a sequence of positive rationals $\overline{\varepsilon} \to 0$ such that for all n:*

1. *if $\mathbb{P}_n(\phi_n) \in (a_n + \delta_n, b_n - \delta_n)$, then*

$$\mathbb{P}_n(\text{“}\underline{a_n} < \underline{\mathbb{P}_n}(\underline{\phi_n}) < \underline{b_n}\text{”}) > 1 - \varepsilon_n,$$

2. *if $\mathbb{P}_n(\phi_n) \notin (a_n - \delta_n, b_n + \delta_n)$, then*

$$\mathbb{P}_n(\text{“}\underline{a_n} < \underline{\mathbb{P}_n}(\underline{\phi_n}) < \underline{b_n}\text{”}) < \varepsilon_n.$$

In other words, for any pattern in $\overline{\mathbb{P}}$'s beliefs that can be efficiently written down (such as "$\overline{\mathbb{P}}$'s probabilities on $\overline{\phi}$ are between a and b on these days"), $\overline{\mathbb{P}}$ learns to believe the pattern if it's true, and to disbelieve it if it's false (with vanishing error). (Recall that the underlines indicate that the underlined expression should be expanded to the appropriate logical formula or term, representing e.g., the source code of an algorithm implementing $\overline{\mathbb{P}}$.)

At a first glance, this sort of self-reflection may seem to make logical inductors vulnerable to paradox. For example, consider the sequence of sentences $\overline{\chi^{0.5}}$ defined using the diagonal lemma by

$$\chi_n^{0.5} := \text{“}\underline{\mathbb{P}_n}(\underline{\chi_n^{0.5}}) < 0.5\text{”}$$

such that $\chi_n^{0.5}$ is true iff $\overline{\mathbb{P}}$ assigns it a probability less than 50% on day n. Such a sequence can be defined by Gödel's diagonal lemma. These sentences are probabilistic versions of the classic "liar sentence", which has caused quite a ruckus in the setting of formal logic [24, 39, 20, 25, 11]. Because our setting is probabilistic, it's perhaps most closely related to the "unexpected hanging" paradox—$\chi_n^{0.5}$ is true iff $\overline{\mathbb{P}}$ thinks it is unlikely on day n. How do logical inductors handle this sort of paradox?

Theorem 5.4.2 (Paradox Resistance)**.** *Fix a rational $p \in (0,1)$, and define an e.c. sequence of "paradoxical sentences" $\overline{\chi^p}$ satisfying*

$$\Gamma \vdash \underline{\chi_n^p} \leftrightarrow \left(\underline{\mathbb{P}_n}(\underline{\chi_n^p}) < \underline{p}\right)$$

for all n. Then

$$\lim_{n\to\infty} \mathbb{P}_n(\chi_n^p) = p.$$

In words, a logical inductor responds to paradoxical sentences $\overline{\chi^p}$ by assigning them probabilities that converge on p.

To understand why this is desirable, imagine that your friend owns a high-precision brain-scanner and can read off your beliefs. Imagine they ask you what probability you assign to the claim "you will assign probability $<80\%$ to this claim at precisely 10am tomorrow". As 10am approaches, what happens to your belief in this claim? If you become extremely confident that it's going to be true, then your confidence should drop. But if you become fairly confident it's going to be false, then your confidence should spike. Thus, your probabilities should oscillate, pushing your belief so close to 80% that you're not quite sure which way the brain scanner will actually call the claim, and you think the scanner is roughly 80% likely to call it true. In response to a paradoxical claim, this is exactly how $\overline{\mathbb{P}}$ behaves, once it's learned how the paradoxical sentences work.

5.5 Self-Trust

We've seen that logical inductors can, without paradox, have accurate beliefs about their own current beliefs. Next, we turn our attention to the question of what a logical inductor believes about its *future* beliefs.

The coherence conditions of classical probability theory guarantee that, though a probabilistic reasoner expects their future beliefs to change in response to new empirical observations, they don't e.g., believe that their future credence in ϕ is, in net expectation, lower than their current credence in ϕ. For example, if a reasoner $\Pr(-)$ knows that tomorrow they'll see some evidence e that will convince them that Miss Scarlet was the murderer, then they already believe that she was the murderer today:

$$\Pr(\text{Scarlet}) = \Pr(\text{Scarlet} \mid e)\Pr(e) + \Pr(\text{Scarlet} \mid \neg e)\Pr(\neg e).$$

In colloquial terms, this says "my current beliefs are *already* a mixture of my expected future beliefs, weighted by the probability of the evidence that I expect to see."

Logical inductors obey similar coherence conditions with respect to their future beliefs, with the difference being that a logical inductor updates its belief by gaining more knowledge about *logical* facts, both by observing an ongoing process of deduction and by thinking for longer periods of time.

To refer to $\overline{\mathbb{P}}$'s *expectations* about its future self, we need a notion of logically uncertain variables. To avoid needless detail, suffice it to say that logically determined quantities, such as the output of a given computer program, can be represented and manipulated analogously to random variables in probability theory. We can write these variables as terms representing their value; for example, the variable written "$\underline{\mathbb{P}_n}(\underline{\phi})$" represents the probability assigned to ϕ by $\overline{\mathbb{P}}$ on day n. Using the beliefs $\mathbb{P}_n$ of $\overline{\mathbb{P}}$ about X on day n, we can define the (approximate) expectation $\mathbb{E}_n(X)$.

We also need to know which future self our logical inductor will defer to:

Definition 5.5.1 (Deferral Function)**.** *A function $f : \mathbb{N}^+ \to \mathbb{N}^+$ is called a **deferral function** if*

1. *$f(n) > n$ for all n, and*
2. *as a function of n, $f(n)$ can be computed in time polynomial in $f(n)$.*

Now we can state the sense in which logical inductors don't expect, on net, their future beliefs to be wrong in any particular direction.

Theorem 5.5.2 (No Expected Net Update)**.** *Let f be a deferral function, and let $\overline{\phi}$ be an e.c. sequence of sentences. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n \mathbb{E}_n(\text{“}\underline{\mathbb{P}_{f(\underline{n})}}(\underline{\phi_n})\text{”}).$$

This theorem only says that $\mathbb{P}_n$ doesn't expect the beliefs of $\mathbb{P}_{f(n)}$ about $\overline{\phi}$ to err in a particular direction. A priori, it is possible that $\mathbb{P}_n$ nevertheless believes its future beliefs $\mathbb{P}_{f(n)}$ will be grossly misguided. For example, suppose that $\mathbb{P}_n$ is very confident that $\mathbb{P}_{f(n)}$ will have sufficient time to compute the truth of ϕ, but will react insanely to this information:

$$\mathbb{P}_n(\text{“}\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi}) = 0\text{”} \mid \phi) = 1$$

and

$$\mathbb{P}_n(\text{“}\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi}) = 1\text{”} \mid \neg\phi) = 1.$$

This is a priori consistent with Theorem 5.5.2 so long as $\mathbb{P}_n$ assigns $\mathbb{P}_n(\phi) = 0.5$, but it clearly indicates that $\mathbb{P}_n$ does not trust its future beliefs.

To instead formalize the idea of a reasoner Pr that trusts their own reasoning process, let us first consider a self-trust property in the setting of deductive logic:

$$\vdash \Box\phi \to \phi.$$

This property of deductive systems says that the system proves “If I prove ϕ at some point, then it is true”. However, any sufficiently strong reasoner that satisfies this property for the statement $\phi = \bot$ is inconsistent by Gödel's second incompleteness theorem! The search for logics that place confidence in their own machinery dates at least back to Hilbert [29]. While Gödel et al. [22] dashed these hopes, it is still desirable for reasoners to trust their reasoning process relatively well, most of the time (which humans seem to do).

As discussed in Section 5.3, logical inductors trust their underlying deductive process $\overline{D}$ in a slightly weaker, finitary sense. More interestingly, it turns out that logical inductors also trust their own reasoning process as a whole, including their inductive conclusions, in a manner that we now formalize.

Instead of $\vdash \Box\phi \to \phi$, we can replace provability with high confidence, and then ask for something like

$$\mathrm{Pr}_{\mathrm{now}}(\phi \mid \mathrm{Pr}_{\mathrm{later}}(\phi) > p) \gtrsim p.$$

Colloquially, this says that if we tell Pr that in the future they will place more than p credence in ϕ, then they update their current beliefs to place at least p credence. In short, Pr trusts that the outputs of their own ongoing reasoning process will be accurate.

Now, in fact property 5.5 is not quite desirable as stated (and logical inductors do not satisfy it). Indeed, consider the liar sentence χ^p defined by

$$\chi^p := \text{“}\mathrm{Pr}_{\mathrm{later}}(\chi^p) < p\text{”}.$$

A good reasoner will then satisfy

$$\mathrm{Pr}_{\mathrm{now}}(\chi^p \mid \mathrm{Pr}_{\mathrm{later}}(\chi^p) > p) \eqsim 0,$$

contradicting equation 5.5. The issue is that if we give $\mathrm{Pr}_{\mathrm{now}}$ high-precision access to the probabilities assigned by $\mathrm{Pr}_{\mathrm{later}}$—for example by conditioning on them—then $\mathrm{Pr}_{\mathrm{now}}$ can outperform the (unconditioned) beliefs of $\mathrm{Pr}_{\mathrm{later}}$, in this case by having correct opinions about the liar sentence for $\mathrm{Pr}_{\mathrm{later}}$.

Instead, we have the following self-trust property, which only gives $\mathbb{P}_n$ limited-precision access to the beliefs of $\mathbb{P}_{f(n)}$:

Theorem 5.5.3 (Self-Trust)**.** *Let f be a deferral function, $\overline{\phi}$ be an e.c. sequence of sentences, $\overline{\delta}$ be an e.c. sequence of positive rational numbers, and $\overline{p}$ be an e.c. sequence of rational probabilities. Then*

$$\mathbb{E}_n\left(\text{“}\underline{\mathbb{1}(\phi_n)}\cdot\underline{\text{Ind}_{\delta_n}}\left(\underline{\mathbb{P}_{f(n)}}(\underline{\phi_n})>\underline{p_n}\right)\text{”}\right) \gtrsim_n p_n\cdot\mathbb{E}_n\left(\text{“}\underline{\text{Ind}_{\delta_n}}\left(\underline{\mathbb{P}_{f(n)}}(\underline{\phi_n})>\underline{p_n}\right)\text{”}\right).$$

The indicator variable $\mathbb{1}(\phi)$ represents 1 if ϕ is true and 0 if ϕ is false. The continuous indicator variable $\text{Ind}_\delta(X>p)$ is an ordinary indicator of the event $X>p$, except that instead of a discontinuity at $X=p$, the value is linear in X on a region of length δ. Thus the self-trust property gives $\mathbb{P}_n$ only continuous (limited precision) access to the beliefs of $\mathbb{P}_{f(n)}$; except for this subtlety, we could have written the more recognizable (but false and undesirable!) statement

$$\mathbb{P}_n\left(\text{“}\underline{\phi_n}\wedge\left(\underline{\mathbb{P}_{f(n)}}(\underline{\phi_n})>\underline{p_n}\right)\text{”}\right)\gtrsim_n p_n\cdot\mathbb{P}_n\left(\text{“}\underline{\mathbb{P}_{f(n)}}(\underline{\phi_n})>\underline{p_n}\text{”}\right),$$

where the conditional $\mathbb{P}_n\left(\text{“}\underline{\phi_n}\mid\underline{\mathbb{P}_{f(n)}}(\underline{\phi_n})>\underline{p_n}\text{”}\right)$ has been rearranged to avoid a potential division by 0.

6 Discussion

We have proposed the *logical induction criterion* as a criterion on the beliefs of deductively limited reasoners, and we have described how reasoners who satisfy this criterion (*logical inductors*) possess many desirable properties when it comes to developing beliefs about logical statements (including statements about mathematical facts, long-running computations, and the reasoner themself).

That said, there are clear drawbacks to the logical inductor we describe in [18]: it does not use its resources efficiently; it is not a decision-making algorithm (i.e., it does not “think about what to think about”); and the properties above hold either asymptotically (with poor convergence bounds) or in the limit. Further, it is unclear whether logical inductors have good beliefs about counterpossibilities, and whether they take advantage of old evidence. These are enticing directions for further research.

The authors are particularly interested in tools that help AI scientists attain novel statistical guarantees in settings where robustness and reliability guarantees are currently difficult to come by. For example, consider the task of designing an AI system that reasons about the behavior of computer programs, or that reasons about its own beliefs and its own effects on the world. While practical algorithms for achieving these feats are sure to make use of heuristics and approximations, we believe scientists will have an easier time designing robust and reliable systems if they have some way to relate those approximations to theoretical algorithms that are known to behave well in principle. Modern models of rational behavior are not up to this task: formal logic is inadequate when it comes to modeling self-reference, and probability theory is inadequate when it comes to modeling logical uncertainty. We see logical induction as a first step towards models of rational behavior that work in settings where agents must reason about themselves, while deductively limited.

6.1 Acknowledgements

We acknowledge Abram Demski, Benya Fallenstein, Daniel Filan, Eliezer Yudkowsky, Jan Leike, János Kramár, Nisan Stiennon, Patrick LaVictoire, Paul Christiano, Sam Eisenstat, Scott Aaronson, and Vadim Kosoy, for valuable comments and discussions. We also acknowledge contributions from attendees of the MIRI summer fellows program, the MIRIxLA group, and the MIRIχ group.

This research was supported as part of the Future of Life Institute (futureoflife.org) FLI-RFP-AI1 program, grant #2015-144576.

References

[1] Scott Aaronson (2013): *Why Philosophers Should Care About Computational Complexity*. In B. Jack Copeland, Carl J. Posy & Oron Shagrir, editors: *Computability: Turing, Gödel, Church, and Beyond*, MIT Press. Available at https://arxiv.org/abs/1108.1791.

[2] Ernest W. Adams (1996): *A Primer of Probability Logic*. University of Chicago Press.

[3] Philippe Balbiani, David Fernández-Duque & Emiliano Lorini (2016): *A Logical Theory of Belief Dynamics for Resource-Bounded Agents*. In: *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems, pp. 644–652. Available at http://dl.acm.org/citation.cfm?id=2936924.2937020.

[4] George Boole (1854): *An Investigation of the Laws of Thought: On which are Founded the Mathematical Theories of Logic and Probabilities*. Dover Publications.

[5] Catrin Campbell-Moore (2015): *How to Express Self-Referential Probability. A Kripkean Proposal*. *The Review of Symbolic Logic* 8(04), pp. 680–704, doi:10.1017/S1755020315000118.

[6] Rudolf Carnap (1962): *Logical Foundations of Probability*. University of Chicago Press, doi:10.2307/2021419.

[7] Paul Christiano (2014): *Non-Omniscience, Probabilistic Inference, and Metamathematics*. Technical Report 2014–3, Machine Intelligence Research Institute. Available at http://intelligence.org/files/Non-Omniscience.pdf.

[8] Paul Christiano, Eliezer Yudkowsky, Marcello Herreshoff & Mihály Bárász (2013): *Definability of Truth in Probabilistic Logic*. Technical Report, Machine Intelligence Research Institute. Available at https://intelligence.org/files/DefinabilityTruthDraft.pdf.

[9] Abram Demski (2012): *Logical Prior Probability*. *Artificial General Intelligence. 5th International Conference, AGI 2012* (7716), pp. 50–59, doi:10.1007/978-3-642-35506-6_6.

[10] Ellery Eells (1990): *Bayesian Problems of Old Evidence*. *Scientific Theories* 14, pp. 205–223. Available at http://mcps.umn.edu/philosophy/14_9Eells.pdf.

[11] Matti Eklund (2002): *Inconsistent Languages*. *Philosophy and Phenomenological Research* 64, pp. 251–275, doi:10.1111/j.1933-1592.2002.tb00001.x.

[12] Ronald Fagin & Joseph Y. Halpern (1987): *Belief, Awareness, and Limited Reasoning*. *Artificial Intelligence* 34(1), pp. 39–76, doi:10.1016/0004-3702(87)90003-8.

[13] Ronald Fagin, Joseph Y. Halpern, Yoram Moses & Moshe Vardi (1995): *Reasoning about Knowledge*. MIT Press Cambridge.

[14] Bruno de Finetti (1937): *Foresight: Its Logical Laws, Its Subjective Sources*. In Henry E. Kyburg & Howard E.K. Smokler, editors: *Studies in Subjective Probability*, Roger E. Kreiger Publishing Co., doi:10.1007/978-1-4612-0919-5_10.

[15] Haim Gaifman (1964): *Concerning Measures in First Order Calculi*. *Israel Journal of Mathematics* 2(1), pp. 1–18.

[16] Haim Gaifman & Marc Snir (1982): *Probabilities over Rich Languages, Testing and Randomness*. *Journal of Symbolic Logic* 47(03), pp. 495–548, doi:10.2307/2273587.

[17] Daniel Garber (1983): *Old Evidence and Logical Omniscience in Bayesian Confirmation Theory*. *Testing scientific theories* 10, pp. 99–131.

[18] Scott Garrabrant, Tsvi Benson-Tilsen, Andrew Critch, Nate Soares & Jessica Taylor (2016): *Logical Induction*. *arXiv:1609.03543 [cs.AI]*. Available at https://arxiv.org/abs/1609.03543.

[19] Scott Garrabrant, Benya Fallenstein, Abram Demski & Nate Soares (2016): *Inductive Coherence*. *arXiv:1604.05288 [cs.AI]*. Available at https://arxiv.org/abs/1604.05288.

[20] Michael Glanzberg (2001): *The Liar in Context*. *Philosophical Studies* 103(3), pp. 217–251, doi:10.1023/A:1010314719817.

[21] Clark Glymour (1980): *Theory and Evidence*. Princeton University Press.

[22] Kurt Gödel, Stephen Cole Kleene & John Barkley Rosser (1934): *On Undecidable Propositions of Formal Mathematical Systems*. Institute for Advanced Study.

[23] Irving J. Good (1950): *Probability and the Weighing of Evidence*. Charles Griffin, London, doi:10.1017/S0031819100026863.

[24] Patrick Grim (1991): *The Incomplete Universe: Totality, Knowledge, and Truth*. MIT Press.

[25] Anil Gupta & Nuel D. Belnap (1993): *The Revision Theory of Truth*. MIT Press.

[26] Ian Hacking (1967): *Slightly More Realistic Personal Probability*. *Philosophy of Science* 34(4), pp. 311–325, doi:10.1086/288169.

[27] Theodore Hailperin (1996): *Sentential Probability Logic*. Lehigh University Press.

[28] Joseph Y. Halpern (2003): *Reasoning about Uncertainty*. MIT Press.

[29] David Hilbert (1902): *Mathematical Problems*. *Bulletin of the American Mathematical Society* 8(10), pp. 437–480. Available at http://www.ams.org/journals/bull/1902-08-10/S0002-9904-1902-00923-3/S0002-9904-1902-00923-3.pdf.

[30] Jaakko Hintikka (1962): *Knowledge and Belief: An Introduction to the Logic of the Two Notions*. Cornell University Press, doi:10.2307/2271621.

[31] Marcus Hutter, John W. Lloyd, Kee Siong Ng & William T. B. Uther (2013): *Probabilities on Sentences in an Expressive Logic*. *Journal of Applied Logic* 11(4), pp. 386–420, doi:10.1016/j.jal.2013.03.003.

[32] E. T. Jaynes (2003): *Probability Theory*. Cambridge University Press, doi:10.1017/CBO9780511790423.

[33] James M. Joyce (1999): *The Foundations of Causal Decision Theory*. Cambridge Studies in Probability, Induction and Decision Theory, Cambridge University Press, doi:10.1017/CBO9780511498497.

[34] A.N. Kolmogorov (1950): *Foundations of the Theory of Probability*.

[35] Kurt Konolige (1983): *A Deductive Model of Belief*. In: *IJCAI*, 83, pp. 377–381. Available at https://www.ijcai.org/Proceedings/83-1/Papers/090.pdf.

[36] David Lewis (1999): *Papers in Metaphysics and Epistemology*. 2, Cambridge University Press, doi:10.1017/CBO9780511625343.

[37] Ming Li & Paul M. B. Vitányi (1993): *An Introduction to Kolmogorov Complexity and its Applications*, 1 edition. Springer, doi:10.1007/978-1-4757-3860-5.

[38] Jerzy Łoś (1955): *On the Axiomatic Treatment of Probability*. Colloquium Mathematicae 3(2), pp. 125–137. Available at http://eudml.org/doc/209996.

[39] Vann McGee (1990): *Truth, Vagueness, and Paradox: An Essay on the Logic of Truth*. Hackett Publishing.

[40] John von Neumann & Oskar Morgenstern (1944): *Theory of Games and Economic Behavior*, 1st edition. Princeton University Press.

[41] George Polya (1990): *Mathematics and Plausible Reasoning: Patterns of Plausible Inference*. 2, Princeton University Press.

[42] Frank Plumpton Ramsey (1931): *Truth and Probability*. In Richard Bevan Braithwaite, editor: The Foundations of Mathematics and other Logical Essays, Harcourt, Brace, pp. 156–198, doi:10.1007/978-3-319-20451-2_3.

[43] Leonard J Savage (1954): *The Foundations of Statistics*. doi:10.1002/nav.3800010316.

[44] Leonard J Savage (1967): *Difficulties in the theory of personal probability*. Philosophy of Science 34(4), pp. 305–310. Available at http://www.jstor.org/stable/186119.

[45] Ray J. Solomonoff (1964): *A Formal Theory of Inductive Inference. Part I*. Information and Control 7(1), pp. 1–22, doi:10.1016/S0019-9958(64)90223-2.

[46] Ray J. Solomonoff (1964): *A Formal Theory of Inductive Inference. Part II*. Information and Control 7(2), pp. 224–254, doi:10.1016/S0019-9958(64)90131-7.

[47] Jan Sprenger (2015): *A Novel Solution to the Problem of Old Evidence*. Philosophy of Science 82(3), pp. 383–401, doi:10.1086/681767.

[48] Paul Teller (1973): *Conditionalization and Observation*. Synthese 26(2), pp. 218–258, doi:10.1007/978-94-010-1853-1_9.

[49] Alan M. Turing (1936): *On Computable Numbers, with an Application to the Entscheidungsproblem*. Proceedings of the London Mathematical Society 42(230–265), doi:10.1112/plms/s2-42.1.230.

[50] Fernando R Velázquez-Quesada (2014): *Dynamic Epistemic Logic for Implicit and Explicit Beliefs*. Journal of Logic, Language and Information 23(2), pp. 107–140, doi:10.1007/s10849-014-9193-0.

[51] Yitang Zhang (2014): *Bounded Gaps between Primes*. Annals of Mathematics 179(3), pp. 1121–1174, doi:10.4007/annals.2014.179.3.7.

[52] Alexander K. Zvonkin & Leonid A. Levin (1970): *The Complexity of Finite Objects and the Development of the Concepts of Information and Randomness by Means of the Theory of Algorithms*. Russian Mathematical Surveys 25(6), pp. 83–124, doi:10.1070/RM1970v025n06ABEH001269.

[53] Lyle Zynda (1995): *Old Evidence and New Theories*. Philosophical Studies 77(1), pp. 67–95, doi:10.1007/BF00996312.