Logical Induction

Scott Garrabrant, Tsvi Benson-Tilsen, Andrew Critch
Nate Soares, Jessica Taylor

(scott|tsvi|critch|nate|jessica)@intelligence.org

Machine Intelligence Research Institute
http://intelligence.org/

Outline

A very rough plan for this talk:

[10 mins] The problem of logical induction

[50 mins] Technical results

[20 mins] Implications and take-aways

Credences should change with time spent thinking / computing:

	1 min	1 day	∞
#1. $P(D_{10} = 7)$	10%	10%	10%
#2. $P(D_{10} = 7 \mid \text{snapshot})$	10%	15%	16%
#3. $P(10^{th}$ digit of $\sqrt{(10)} = 7)$	10%	1%	0%

Probability theory gives rules for how probabilities should relate to each other and change with new observations, *assuming logical omniscience…*

…but what rules should credences follow over time, as computation is carried out on observations that have already been made?

Also, 50% would be a worse answer to start with here... can we make a principled theory from which this claim would follow?

snapshot for #2:

Goal: call the purple processes "**logical induction**" and figure out how it should work.

Past desiderata for “good reasoning” under logical uncertainty:

1. **computable approximability** — the process should be approximable by a Turing Machine. (Demsky, 2012)
2. **coherent limit** — after infinite time, credences should satisfy the laws of probability theory, such as $(A \rightarrow B) \Rightarrow (P(A) \leq P(B))$. (Gaifman, 1964).
3. **partial coherence:** credences at finites time should roughly satisfy some coherence properties; such as $Q(A \wedge B) + Q(A \vee B) \approx Q(A) + Q(B)$ (Good, 1950; Hacking, 1967)
4. **calibration** — the process should be right roughly 90% of the time when it's 90% confident. (Savage, 1967)
5. **introspection** — the process should be able to describe and reason about itself. (Hintikka, 1962; Fagin, 1995; Christiano, 2013; Campbell-Moore, 2015)
6. **self-trust** — it should understand that it is reliable and that it will become more reliable with time (Hilbert, 1900)
7. **non-dogmatism** — it does not assign 100% or 0% credence to claims unless they have been proven or disproven, respectively (Carnap, 1962; Gaifman, 1982; Snir, 1982)
8. **PA-capable** — it should assign non-zero probability to the consistency of Peano Arithmetic, i.e. to the set of consistent completions of PA.
9. **rough inexploitability** — it should not be easy to ``dutch book'' the process / make bets against it that are guaranteed to win (von Neumann and Morgenstern 1944; de Finetti 1979)
10. **Gaifman inductivity** — it should come to believe $(\forall x, f(x))$ in the limit as it examines every example of x and confirms f(x) (Gaifman 1964, Hutter 2013)
11. **Efficiency** — it runs in polynomial (preferably quadratic) time
12. **Decision-relevant** — should be able to focus computation on questions relevant to decisions.
13. **Updates on old evidence** (Glymour, 1980)

Why develop a theoretical model of logical induction?

One motivation is to help us reason about highly capable AI systems before they exist. Without a source code in hand, we tend to fall back to thinking of advanced systems as being "good at stuff", like:

choosing actions to achieve objectives given beliefs
→ it roughly obeys **rational choice** theory (e.g. VNM theorem)

updating beliefs according to new evidence
→ it roughly obeys **probability** theory (e.g. Bayes' theorem)

computing belief updates with resource limitations
→ it roughly obeys **<?????>** theory (e.g. <*****> theorem)

In hopes of developing it, **<?????>** has been called "**logical uncertainty**", and we call the process of refining logical uncertainties "**logical induction**".

Let's defer further questions until the idea has been made more precise; for now just remember that logical induction is about what beliefs should look like before computations are finished:

	1 min	1 day	∞
#1. $P(D_{10} = 7)$	10%	10%	10%
#2. $P(D_{10} = 7 \mid \text{snapshot})$	10% →	15% →	16%
#3. $P(10^{th}$ digit of $\sqrt{(10)} = 7)$	10%	1%	0%

Formalizing logical induction

PowerPoint → Beamer

Formalizing logical induction

Beamer → PowerPoint

The current state of logical uncertainty theory

Domain of Study	Agent Concept	Minimalistic Sufficient Conditions	Desirability Arguments	Feasibility
rational choice theory / economics	VNM utility maximizer	VNM axioms	Dutch book arguments, compelling axioms, …	AIXI, POMDP solvers, …
probability theory	Bayesian updater	axioms of probability theory	Dutch book arguments, compelling axioms, …	Solomonoff induction
logical uncertainty theory	**Garrabrant inductor**	???	**Dutch book arguments, historical desiderata, …**	**LIA2016**

recent progress

What have we learned so far?

The following are more feasible than one might think:

- **Inexploitability.** An algorithm can satisfy a fairly arbitrary set of inexploitability conditions using Brouwer's FPT.
- **Self-trust.** Introspection and self-trust need not lead to mathematical paradoxes.
- **Outpacing deduction.** Inductive learning can in principle outpace deduction, by an uncomputably large margin on efficiently computable questions.

What have we learned so far?

The following are less "required" than one might think for a rational gambler to avoid exploitation:

- **Calibration.** So far it looks like one need only be calibrated about sequences of logical bets that are settled sufficiently quickly (this is being actively researched).

- **Hard-coded belief coherence.** A powerful bet-balancing procedure can and must learn to "mimic" deductive rules used to settles its bets.

Paths forward

1. **Improving** logical inductor theory (Minimalistic conditions? Mutual dominance? Other open questions...)

2. **Using** Garrabrant inductors / LIA2016 to ask new questions about AI alignment

3. **Other approaches** to AI alignment*

MIRI's focus

* Must eventually address logical uncertainty implicitly or explicitly, so expect some convergence.

How will logical induction be applicable?

Conceptual tools for reasoning about **incentives, competition, and goal pursuit** are under-developed for computationally bounded agents. They presume agents are logically omniscient, because we already had good theoretical models for developing them that way:

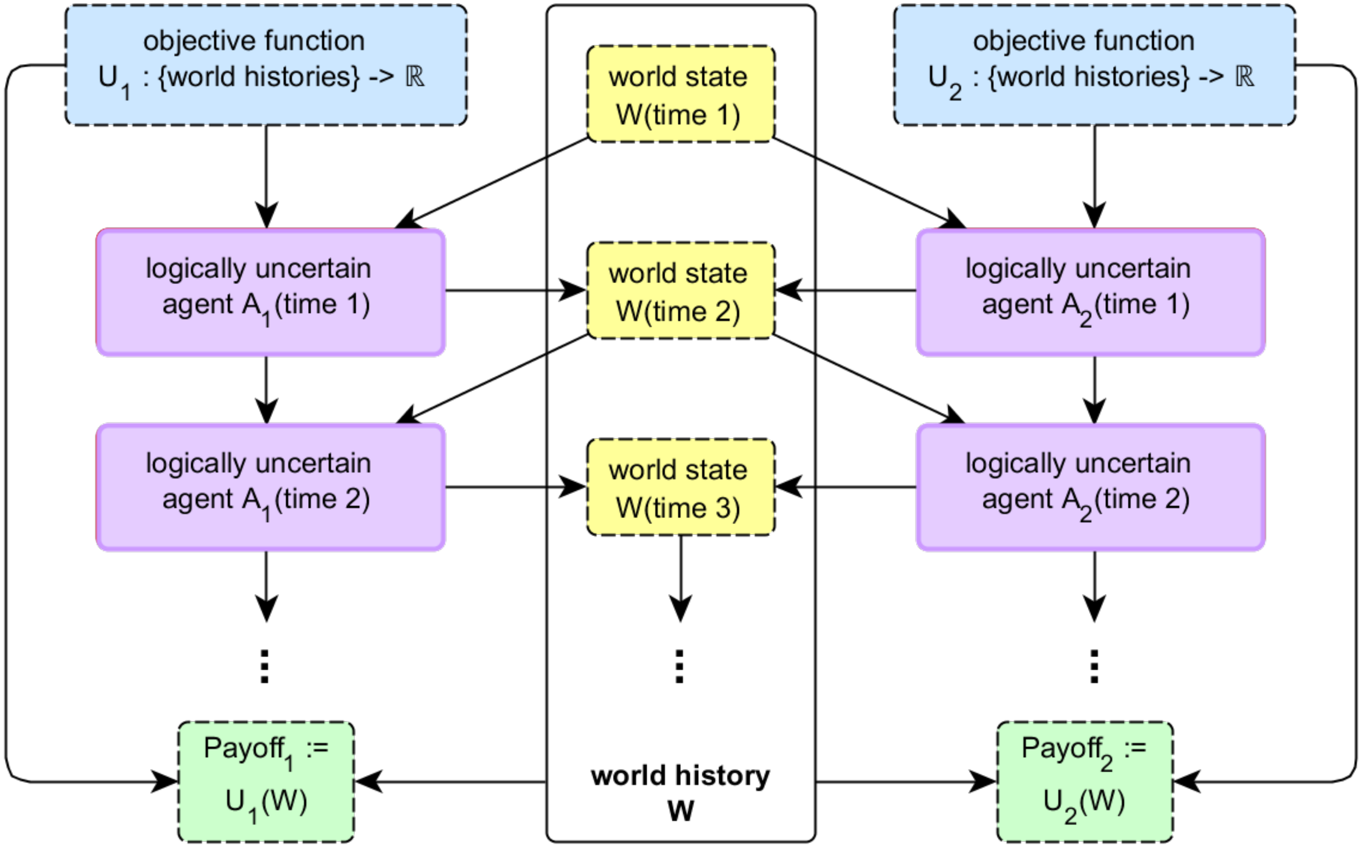
- **Game theory and economics:**
  - Von Neumann-Morgenstern utility theorem
  - Nash equilibria and correlated equilibria
  - Efficient market theory:
    - Fundamental theorems of welfare economics
    - Coase's Theorem
  - Value of Information (VOI)
- **Mechanism design**
  - Gibbard–Satterthwaite theorem
  - Myerson–Satterthwaite theorem
  - Revenue Equivalence theorem

Theoretical models of limited (and eventually, bounded) reasoners could help expand these fields to ask more questions directly relevant to artificial agents.

Visualizing a theoretical application

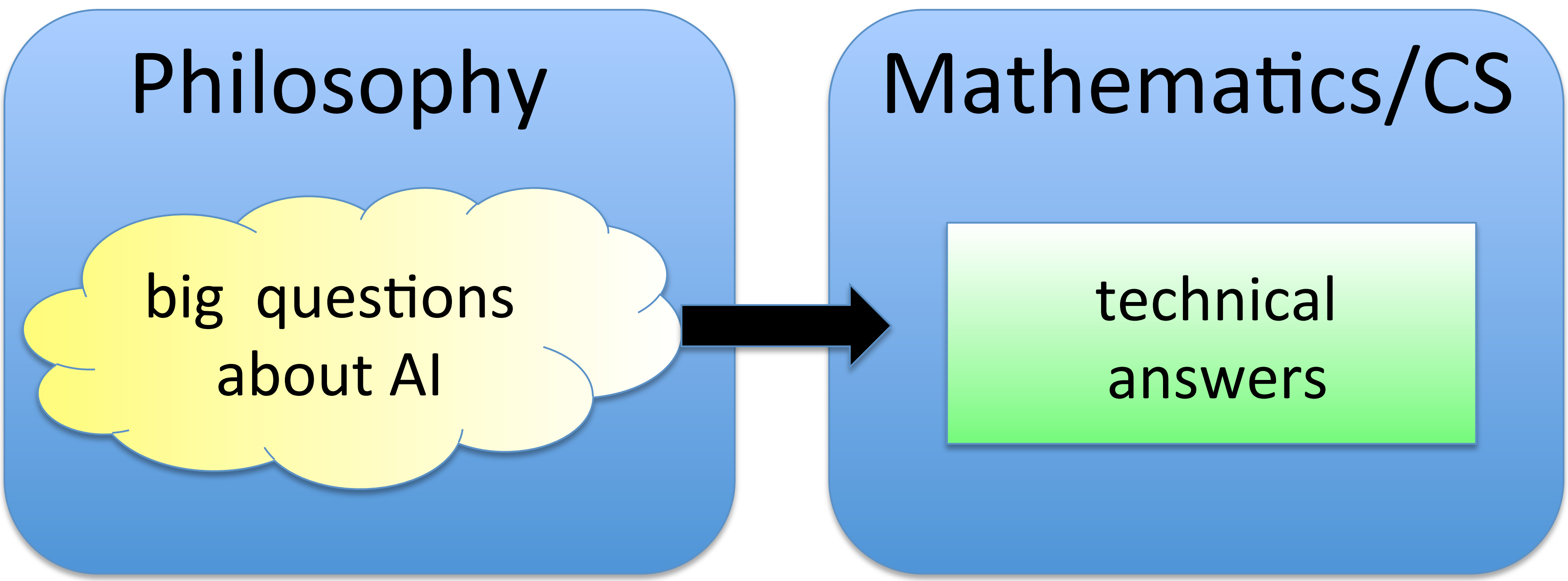
Currently, game theory analyzes scenarios with logically omniscient agents…

Now we can better theoretically analyze scenarios with bounded reasoners:



Meta updates

MIRI's general approach includes developing "big" questions about how AI can and should work, past the stages of philosophical conversation and into the domain of math and CS.



Meta updates

I was not personally expecting logical induction to be “solved” in this way for at least a decade, so I’ve updated that:

- I would like to see more theoreticians trying to beak down unsettled philosophical questions about intelligence and AI into math/CS and grinding through them like this; and
- perhaps other seemingly “out of reach” problems in AI alignment, like decision theory and logical counterfactuals, might be amenable to this sort of approach.

Thanks!

To

- **Scott Garrabrant**, for the core idea and many rapid subsequent insights;
- **Tsvi Benson Tilsen**, **Nate Soares**, and **Jessica Taylor** for co-developing the theory and resulting paper; and
- **Jimmy Rintjema** for a *lot* of help with LaTeX bugs and collaborative editing issues

<end of this talk>