Limit-Computable Grains of Truth for Arbitrary Computable Extensive-Form (Un)Known Games

Cole Wyeth[1], Marcus Hutter[2,3], Jan Leike[4], and Jessica Taylor[5]

[1]David R. Cheriton School of Computer Science, University of Waterloo
[2]Google DeepMind
[3]School of Computing, Australian National University
[4]Anthropic
[5]Median Group

September 11, 2024

Abstract

A Bayesian agent acting in a multi-agent environment learns to predict the other agents' policies if its prior assigns positive probability to them (in other words, its prior contains a *grain of truth*). Finding a reasonably large class of policies that contains the Bayes-optimal policies with respect to this class is known as the *grain of truth problem*. Only small classes are known to have a grain of truth and the literature contains several related impossibility results. In this paper we present a formal and general solution to the full grain of truth problem: we construct a class of policies that contains all computable policies as well as Bayes-optimal policies for every lower semicomputable prior over the class. When the environment is a known repeated stage game, we show convergence in the sense of [KL93a] and [KL93b]. When the environment is unknown, Bayes-optimal agents may fail to act optimally even asymptotically. However, agents based on Thompson sampling converge to play ε-Nash equilibria in arbitrary unknown computable multi-agent environments. Finally, we include an application to self-predictive policies that avoid planning. While these results are purely theoretical, we show that they can be computationally approximated arbitrarily closely.

1 Introduction

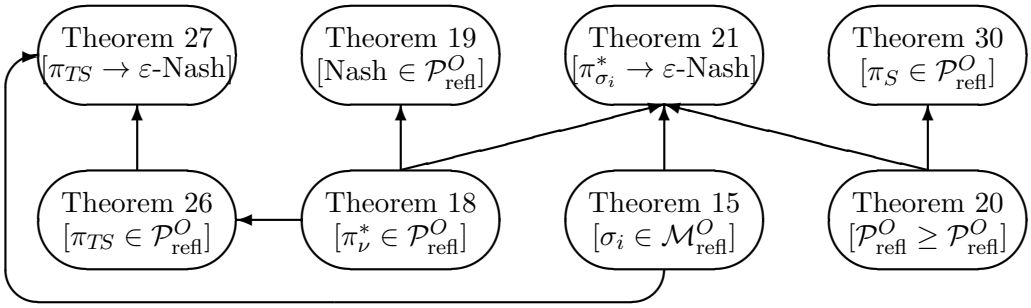
We will consider the behavior of Bayesian players engaged in infinite games. This is a core problem of game theory, but because of reflective difficulties the first rich class of solutions grounded in rigorous decision theory has only recently been proposed [LTF16]. According to the standards of rational behavior derived by von Neumann and Morgenstern [NMR44], players should act to maximize their expected utility. This requires each individual subject among the players to maintain and update beliefs about all other players' strategies, which presumably depend on their reasoning about the subject's strategy. This well studied infinite recursion gives rise to the grain of truth problem [KL93a]: how can one construct a consistent set of beliefs for each player such that they all assign nonzero probability to each others' Bayes-optimal strategies? This problem is known to be difficult, with many impossibility results (e.g. [Nac97; Nac05; FY01]).

Solution overview. Using the recently invented concept of a "reflective oracle" [FTC15] one can construct priors that overcome this difficulty and are even limit-computable. Intuitively, reflective oracles allow algorithms to make predictions about their own behavior, which would normally be impossible because of diagonalization. We will introduce reflective oracles and show how they can be used to construct a class of probability measures $\mathcal{P}^O_{\text{refl}}$ in Section 4. In Section 5 we show that when strategies are chosen in $\mathcal{P}^O_{\text{refl}}$, each player's "subjective environment" is reflective-oracle computable and they have an optimal policy in $\mathcal{P}^O_{\text{refl}}$ (Theorem 15 and Theorem 18). This allows us to construct an interesting reflective-oracle computable Nash equilibrium in Theorem 19. To model players' uncertainty about the strategies they face we construct a dominant "mixture" policy $\zeta \in \mathcal{P}^O_{\text{refl}}$ in Section 7 Theorem 20, which is the final step to establish the grain of truth property classes in our strict sense. This allows us to satisfy the conditions of [KL93a], proving that Bayesian players with beliefs supported on $\mathcal{P}^O_{\text{refl}}$ will converge to an ε-Nash equilibrum in an infinitely repeated stage game; this is a particularly surprising result because mixed strategy Nash equilbria arise naturally despite the fact that Bayesians are not a priori required to randomize. Finally, we address the general case that players do not know either the game or their opponents' strategies, but only know the classes the game and strategy are drawn from. We establish a grain of truth property for this case showing that Thompson sampling versions of the Bayes-optimal strategies are in the strategy class, so are assigned non-zero probability. This allows us to use single-agent asymptotic optimality results for Thompson sampling in an unknown environment to prove convergence to a ε-Nash equilibrium:

> *There is a class of limit-computable strategies satisfying the grain of truth property with respect to any computable game, and it includes limit-computable strategies that converge to a ε-Nash equilibrium even when the game is unknown.*

A brief outline of the paper including dependencies between major theorems appears below.

Contributions. We solve the long-standing grain of truth problem by introducing a class of reflective-oracle computable strategies. This allows us to establish convergence of Bayesian players to ε-Nash equilibrium in known repeated games, followed by convergence for Thompson sampling strategies on unknown games. The rigor and elegance of proofs are improved over the conference version of this paper [LTF16] by extending reflective oracles to non-binary alphabets with "types" for distinct action and percept spaces. We also include a novel application of the machinery developed for the grain of truth problem to answer a question posed in [Cat+23], constructing a self-predictive agent with consistent beliefs about its own policy and suggesting a direction for further research. All results are shown to be limit-computable.



2 Mathematical Preliminaries

Notation. $\mathcal{A}^*$ is the set of finite strings x over a finite set of alphabet symbols $a \in \mathcal{A}$. We will use $x_t \in \mathcal{A}$ to denote the t^{th} element of such a string (indexing from 1) and $x_{1:t} = x_{\leq t}$ is the substring $x_1 x_2 ... x_t \in \mathcal{A}^*$; $x_{<t}$ and $x_{\neq t}$ are defined analogously. The string xy is the concatenation of x and y. We bars $|\cdot|$ are overloaded, representing length for strings, cardinality for sets, and absolute value for reals. While the index t is reserved for "temporal" indexing such as elements of an ordered string or sequence, we will reserve n for the number of players in a game and $1 \leq i, j \leq n$ for the indices of (respectively current and other) players in the game. We will use $\mathcal{T}$ to denote the set of probabilistic Turing machines with oracle access. We will use Δ to represent a probability simplex; for example $\Delta\mathcal{A}$ is a probability distribution on $\mathcal{A}$. For probability measures μ, ν we write "μ is absolutely continuous w.r.t. ν" as $\mu \ll \nu$. The Iverson brackets $[\![R]\!]$ are 1 when R is true and 0 when R is false (they cast booleans to integers). Further notation will be introduced as needed; see Appendix A for a complete list.

We will need the following computability levels from the arithmetic hierarchy.

Definition 1 (computability) A function f is

- *(finitely) computable* (or recursive) if it is computed by some Turing machine.

- *estimable* if there is a computable function $\phi(x,k)$ such that $\forall k|f(x)-\phi(x,k)|<\frac{1}{k}$. That is, f can be approximated to arbitrary pre-specified precision.
- *lower semicomputable* (l.s.c.) if there is a computable function $\phi(x,k)$ monotonically increasing in its second argument with $\lim_{k\to\infty}\phi(x,k)=f(x)$. That is, f can be approximated from below.
- *limit-computable* (or approximable) if there is a computable function $\phi(x,k)\to f(x)$ for $k\to\infty$. That is, f can be approximated to arbitrary but unknown precision.

An estimatable function is always l.s.c.: If ϕ estimates f, then $\phi'(x,k') := \max_{k\le k'}\{\phi(x,k)-\frac{1}{k}\}$ lower semicomputes $f(x)$. Estimable functions are often called 'computable' but we find it safer to not overload the term.

Definition 2 (semimeasure)
A semimeasure ν is a function $\mathcal{A}^*\to\mathbb{R}^+$ satisfying $\nu(x)\ge\sum_{a\in\mathcal{A}}\nu(xa)$.

For our purposes semimeasures always assign probability 1 to the empty string ε. Because algorithms do not always halt, the objects of algorithmic probability are often semimeasures with probability gaps arising from non-halting behavior. Semimeasures are designed for sequence prediction and assign a (defective) probability to observing a sequence starting with string x. Another viewpoint more in line with measure theory is that x represents the cylinder set Γ_x including all infinite continuations of x. It should be noted that there is not a standard well-defined extension of semimeasures to arbitrary sets of infinite strings analogous to Carathéodory's extension theorem for measures. Constructing such an extension or proving its impossibility is an interesting and apparently open problem [HQC24, Sec.2.8.2].

3 The Grain of Truth Problem

Problem statement. The grain of truth problem concerns a set of n players engaged in a multi-player game σ in some (countable) class of games $\mathcal{G}$. Each player believes that the other players' strategies are drawn independently from a (countable) policy class $\mathcal{P}$. Then it is natural to ask whether under some choices of prior over $\mathcal{P}$, a Bayesian optimal strategy for each player is itself in $\mathcal{P}$. In other words, we are seeking conditions that make the players' beliefs about each other "consistent" and subjectively optimal. We can also view this as a single agent interacting with an environment ν which consists of game σ combined with the remaining agents (in contrast to games, environments have only one "player"). If and only if $(\mathcal{G},\mathcal{P})$ contain a grain of truth in the above sense, the condition that the true environment is $\mu\in\mathcal{M}$ is satisfied. Consider a multi-agent setup where n agents interact with each other via a common environment in rounds. In game-theoretic parlance, player i follows some mixed strategy π_i from some class of strategies $\mathcal{P}$. The extensive-form "game" σ they are "playing" also includes observations and rewards to the agents,

where the utility is the expected discounted reward sum. Conventionally a repeated known game and Nash equilibria are considered. We deal with this case only as a stepping stone to our much more general setting: Our main setting and results consider one long extensive-form game which is only known to belong to a countable class of games $\mathcal{G}$, and essentially no structural assumptions are made on $\mathcal{G}$. We also do not assume that players play Nash equilibria. Player i only assumes that the others' policies $\pi_j \in \mathcal{P}$ and that the game $\sigma \in \mathcal{G}$[1]. From player i's perspective, he interacts with an environment σ_i that consists of game $\sigma \in \mathcal{G}$ and a strategy profile $\pi_{\neq i} := (\pi_1, ..., \pi_{i-1}, \pi_{i+1}, ..., \pi_n) \in \mathcal{P}^{n-1}$ of the other players $j \neq i$. Player i does not know σ and does not assume that $\pi_{\neq i}$ is a Nash strategy, so needs to infer both from the experienced interaction history $h_{<t}$. This setup is more realistic in that it allows to model agents that learn from experience and do not assume any particular strategy (e.g. Nash) of their "opponents" beyond being in $\mathcal{P}$, and do not need to know the game they are playing. This is the multi-agent version of the optimal history-based reinforcement learning agent AIXI [Hut05; HQC24], and it can model very general problems. For instance, player i may face sub-optimal or colluding or cooperative opponents.

A Bayesian learner chooses Bayes-optimal actions w.r.t. the Bayes-mixture over $(\sigma, \pi_{\neq i})$ w.r.t. some prior w over $\mathcal{G} \times \mathcal{P}^{n-1}$. If there is such a Bayes-optimal strategy in $\mathcal{P}$, we say that $\mathcal{G}, \mathcal{P}$ satisfies the grain of truth property with respect to this choice of prior. There are general theorems which show that (a Thompson-sampling version of) the Bayes-optimal strategy π_i^* is asymptotically optimal in the sense that it converges to the optimal (informed) agent who knows the environment (here $\sigma, \pi_{\neq i}$). This single-agent view is asymmetric in that it singles out one (Bayes-optimal) agent i against $n-1$ agents from some class $\mathcal{P}$. A symmetric treatment requires agent i to consider the possibility that the other agents $j \neq i$ are also Bayes-optimal agents π_j^*. This consideration is formally satisfied iff (the Thompson sampling version of) π_j^* is itself in $\mathcal{P}$ with a non-zero prior $w(\pi_j^*) > 0$. We say that $(\mathcal{P}, \mathcal{G}, w)$ contains a *grain of truth* if this condition is satisfied. The grain of truth problem is the question of whether there exist interesting classes $(\mathcal{P}, \mathcal{G})$ that contain a grain of truth.

As a special case, we construct an interesting policy class satisfying the convergence conditions of [KL93a] in known infinitely repeated games. For these purposes, we prove a weak form of the grain of truth property, formally:

Definition 3 (Grain of truth property) Given a class of policies=strategies $\mathcal{P}$ and a class of games $\mathcal{G}$, consider a vector of policies $\pi = (\pi_1, ..., \pi_n) \in \mathcal{P}^n$. Hold out one player i and construct the environment σ_i^π it is interacting with (the game σ together with the other players' $j \neq i$ strategies). Let player i's beliefs about his environment be described by the Bayesian mixture environment ξ_i. Let $\pi_{\xi_i}^*$ be the

[1]In Section 9 the players' belief distributions can be made in a sense even more general, taking a mixture over a class of subjective environments that includes any combination of a game from $\mathcal{G}$ and opponent strategies from $\mathcal{P}$, but need not explicitly draw a distinction between an opponent and any other part of the environment.

optimal response strategy. We say that $(\mathcal{P}, \mathcal{G})$ contains a grain of truth iff $\forall i,\ \forall \pi_{\neq i} \in \mathcal{P},\ \forall \sigma \in \mathcal{G},\ \sigma_i^\pi \ll \xi_i$ and $\pi^*_{\xi_i} \in \mathcal{P}$.

This is the property we will need to show convergence to ϵ-Nash equilibrium in a known game. The condition $\sigma^p i_i \ll \xi_i$ can be satisfied by choosing ξ_i as an explicit Bayesian mixture over $\mathcal{G} \times \mathcal{P}^{n-1}$ with weights w_i, in which case it makes sense to say that $(\mathcal{P}, \mathcal{G}, w)$ satisfies the grain of truth property, but this is not required by Definition 3. For unknown games ($\mathcal{G} \neq \{\sigma\}$) we need a stronger property for convergence:

Definition 4 (Strong grain of truth property) Given a class of policies=strategies $\mathcal{P}$ and a class of games $\mathcal{G}$, let player i's beliefs about his environment be described by the Bayesian mixture ξ_i with weights $w_i(\nu) > 0$ for $\nu \in \mathcal{M}$. Let $\pi^*_{\xi_i}$ be the optimal response strategy. We say that $(\mathcal{P}, \mathcal{G})$ and specifically $(\mathcal{P}, \mathcal{G}, w)$ contains a strong grain of truth iff $\forall i,\ \forall \pi_{\neq i} \in \mathcal{P},\ \forall \sigma \in \mathcal{G},\ \sigma_i^\pi \in \mathcal{M}$ and $\pi^*_{\xi_i} \in \mathcal{P}$.

In fact, we need the "Thompson sampling version" of Definition 4, obtained by replacing $\pi^*_{\xi_i}$ by the Thompson sampling strategy π_{TS} in the final condition. It is clear that the strong grain of truth property is in fact stronger than the grain of truth property; it implies not only absolute continuity but even bounded Radon-Nikodym derivative $d\sigma_i^\pi / d\xi_i \leq w_i(\sigma_i^\pi)^{-1}$.

History of the grain of truth problem. Progress towards discovering rich strategy and game classes satisfying the grain of truth property has been slow. After [KL93a] introduced the grain of truth property in the context of infinitely repeated games and independent strategies, along with a simple prisoner's dilemma example, many impossibility results were proven ([Nac97; Nac05; FY01]). Much later, [FTC15] introduced reflective oracles, laying the groundwork for a solution (but only considering stage games and not the grain of truth property). This work was extended to sequential decision theory by [FST15], but [LTF16] (the conference version of this paper) was the first to solve the grain of truth problem. However, Leike et al. focused on convergence in unknown games which required Thompson sampling strategies instead of the Bayesian strategies of [KL93a]. As a result they did not correctly formulate or prove a solution to Kalai and Lehrer's problem. We provide such a solution along with more elegant and complete proofs of many of Leike et al.'s other results.

Strategies. We can model strategies as a family of measures for each sequence of observations in a player's information set. The probabilities of the first t actions can only depend on observations available before time t. This is sometimes referred to as a "chronological contextual" measure [HQC24]. Typically, we will show that our grain of truth classes $\mathcal{P}$ contain a "dominant" mixture policy $\zeta \in \mathcal{P}$ such that $\forall \pi \in \mathcal{P}$, $\exists c \in \mathbb{R}^+$ satisfying $\zeta \geq c\pi$. In that case, each player may express their belief that the others choose some strategy in $\mathcal{P}$ by modeling their strategies as ζ (this is essentially Kuhn's theorem [Aum64]; see also [Ale]). In the case that each player has a different action set it is necessary to generalize this problem by indexing the policy classes by player as $(\mathcal{P}_i)_{1 \leq i \leq n}$.

Reinforcement learning	Game theory
stochastic policy	mixed strategy
deterministic policy	pure strategy
agent	player
multi-agent environment	infinite extensive-form game
reward	payoff/utility
(finite) history	history
infinite history	path of play

Table 1: Terminology dictionary between reinforcement learning and game theory from [LTF16].

Multi-player games. Formally, a multi-player game is a chronological contextual measure. Given a sequence of action vectors $a_t = (a_t^1, a_t^2, ..., a_t^n) \in \mathcal{A}^n$, a game σ assigns a probability to the sequence of perception vectors $e_t = (e_t^1, e_t^2, ..., e_t^n) = (o_t^1 r_t^1, o_t^2 r_t^2, ..., o_t^n r_t^n) \in \mathcal{E}^n$ including observations o_t^i and rewards $r_t^i \in [0, 1]$. This is written $\sigma(e_{\leq t} || a_{\leq t})$.

Examples. Any Nash equilibrium of a game σ with strategy π_i for each player i is a trivial "solution" to the grain of truth problem with $\mathcal{G} = \{\sigma\}$, $\mathcal{P}_i = \{\pi_i\}$, but this is not very interesting for our purposes because it does not necessarily model learning.

A basic but non-trivial example is discussed [KL93a]; consider an infinitely repeated prisoner's dilemma. In every time step the payoff matrix is as follows, where C means cooperate and D means defect.

	C	D
C	3/4, 3/4	0, 1
D	1, 0	1/4, 1/4

Let the strategy class be $\mathcal{P} = \{g_t\}_{t \in \mathbb{N} \cup \{\infty\}}$ where g_t is a grim trigger strategy that punishes defection by defecting indefinitely, but by default cooperates until time t and defects afterwards. It is fairly easy to see that regardless of a player's prior belief w_t in each g_t, once he or his opponent has defected, any strategy in $\mathcal{P}$ continues to defect indefinitely, so he expects his opponent to certainly defect. The Bayes optimal (and strictly dominant) strategy for him is therefore to defect from that point on, which is itself a grim trigger strategy. Depending on his priors w_t, he may also expect his opponent to very likely defect at time $t_d < \infty$ despite continued cooperation for $t \leq t_d$, in which case his optimal policy may be $g_{t_d - 1}$. Not only does this $(\mathcal{P}, \mathcal{G}) = (\mathcal{P}, \{\sigma\})$ satisfy the strong grain of truth property, $(\mathcal{P}, \mathcal{G}, w)$ satisfies the strong grain of truth property for any choice of w_i supported on $\{\sigma^\pi | \pi_{\neq i} \in \mathcal{P}^{n-1}\}$. The catch is that $\mathcal{G}$ is only a single (known) environment.

For a much larger (non)example, the class containing all strategies naively appears to satisfy the grain of truth property, but in any nontrivial infinite game it is not

countable and certainly has no dominant strategy, so it is usually not possible to define a useful prior over this class.

Action/perception encodings. The settings we will discuss restrict players and games to be represented by probabilistic Turing machines, so that they accept the interaction history on an input tape and stochastically produce an action or observation on the output tape. The symbolic representation of the history on these tapes can become important. We require it to be uniquely decodeable into actions and observations (for instance, by devoting a consistent number of symbols to each action and observation). We will denote the representation of any object s as $\langle s \rangle$ and assume that $\langle a_1^i e_1^i ... a_t^i e_t^i \rangle = \langle a_1^i \rangle \langle e_1^i \rangle ... \langle a_t^i \rangle \langle e_t^i \rangle$. The simplest "highly granular" representation is to use a unique symbol for each $e_t^i \in \mathcal{E}$[2] and a unique symbol for each $a_t^i \in \mathcal{A}$. Conceptually these symbol sets should be disjoint, but it is always possible to determine which is which by indexical position. By the compositionality properties of Turing machines, if a player's opponents have computable strategies and the game is computable, we can construct a Turing machine to simulate both the opponents and the game from his perspective; this forms a computable *subjective environment*.

Using the terminology introduced above, we can rephrase our main result as follows:

Theorem 5 (limit-computable convergence to equilibrium) There are limit-computable (Thompson sampling) strategies $\pi_1, ..., \pi_n$ such that for any computable multi-player game σ and for all $\varepsilon > 0$ and all $i \in \{1, ..., n\}$ the $\sigma^{\pi_{1:n}}$-probability that the policy π_i is an ε-best response converges to 1 as $t \to \infty$.

4 Reflective Oracles

Rational players can face an infinite regress in which each mutually reasons about the other's reasoning. For instance, if each player's strategy is computed by a commonly known Turing machine, it would seem to be rational to run the other players' machines to predict their behavior and choose the utility maximizing response. When the other players' Turing machines halt, this is computable. Unfortunately, if all players attempt this strategy there is mutual recursion as player 1 simulates player 2 simulating player 1 ad infinitum[3]. Classically this interdependence of optimal strategies is resolved by assuming players will choose a Nash equilibrium, but this is not a Bayesian optimality notion because it is not clear why players should learn to play any particular equilibrium strategy. To model the (subjective) uncertainty of Bayesian

[2] A perception can be encoded as a single symbol as long as the observation and reward can be computably extracted.

[3] An amusing example of this behavior appears in William Goldman's "The Princess Bride," when Vizzini attempts to determine which of two cups Westley poisoned by speculating about what Westley will think Vizzini thinks about Westley. This is isomorphic to the game of matching pennies described in Section 7

players, as well as intentionally randomized behavior strategies, we will use probabilistic Turing machines (pTM's). Formally, a probabilistic Turing Machine (pTM) is a Turing Machine with access to both the ordinary input, output, and work tapes and an additional infinite tape initialized with random bits. To cut through the infinite regress and allow players to consistently reason about each others' strategies, we will allow all pTM's access to the same "reflective" oracle. In this section we show how pTM's model probability distributions (such as behavior strategies) and introduce reflective oracles. Combining these two ideas to construct pTM's with reflective oracle access, we establish some basic computability properties that lay the foundations for the rest of the paper.

Tape alphabet. For simplicity, we will begin with input and output tapes having the same alphabet $\mathcal{A}$, but the theory can easily be extended to include differing input and output alphabets.

Sampling from a pTM. Probabilistic Turing machines are interpreted as computing conditional probabilities. For a pTM T, we define $\lambda_T(\alpha|x)$ to be the probability that on input x, T produces the symbol α (and nothing else). Then we can define a semimeasure

$$\lambda_T(x) = \prod_{t=1}^{|x|} \lambda_T(x_t|x_{<t}) \tag{1}$$

This is not in general a proper probability measure because there is some chance that the pTM does not halt or produces an invalid output. Later, we will find an interesting way to use reflective oracles to complete these semimeasures to probability measures.

Oracle access. Oracle access means that the pTM's can write a query on an oracle tape and enter a special state that queries the oracle, with the next transition depending on the oracle's output. We show in Appendix E, Theorem 44 that for any pTM T, λ_T has l.s.c. conditionals. It is possible to invert this construction, and the other direction (Theorem 45) works even when machines have access to oracles. We define $\lambda_T^O(\alpha|x)$ as the probability that the oracle pTM T with access to O returns α on input x. The semimeasure λ_T^O is defined analogously to before. We will use the symbol ν to represent arbitrary semimeasures.

Theorem 6 (l.s.c. semimeasures vs pTM semimeasures)
A semimeasure ν has l.s.c. conditionals *iff* there exists a pTM T such that $\nu = \lambda_T$

Proof. sketch (detailed proof in Appendix E):
($\Leftarrow$) That the conditionals of λ_T are l.s.c. is rather straight-forward from their construction.
($\Rightarrow$) Let $\phi_\alpha(x, k)$ be computable and monotone increasing in k converging to $\nu(\alpha|x)$ for $\alpha \in \mathcal{A} = \{1, ..., d\}$. Consider a pTM T implementing the following procedure: Let

$\Delta_\alpha(k) := \phi_\alpha(x,k) - \phi_\alpha(x,k-1) \geq 0$ with $\phi_\alpha(x,0) := 0$. Then chop

$$\begin{array}{rl} \text{successive intervals} & I_1(1), ...,\ I_d(1),\ I_1(2), ...,\ I_d(2),\ I_1(3), ... \\ \text{of lengths} & \Delta_1(1), ..., \Delta_d(1), \Delta_1(2), ..., \Delta_d(2), \Delta_1(3), ... \end{array}$$

from interval $[0;1)$. All-together these intervals cover $[0; \sum_\alpha \nu(\alpha|x)) \subseteq [0;1)$. Let $\omega_{1:\infty}$ be uniform random bits. Let T output α if $\exists k : [0.\omega_{1:k}; 0.\omega_{1:k} + 2^{-k}) \subseteq \bigcup_{k'=1}^\infty I_\alpha(k')$. For $0.\omega < \sum_\alpha \nu(\alpha|x)$, the condition can be tested effectively by running through $k = 1, 2, 3, ...$ while only finitely many k' need to be checked. The procedure terminates in finite time, since the interval on the l.h.s. shrinks to a point $(0.\omega)$ for $k \to \infty$, hence eventually is contained in some $I_\alpha(k')$. This procedure outputs α with probability $\nu(\alpha|x)$, since

$$P[0.\omega \in \textstyle\bigcup_{k'=1}^\infty I_\alpha(k')] \ = \ |\bigcup_{k'=1}^\infty I_\alpha(k')| \ = \ \lim_{k\to\infty} \sum_{k'=1}^k \Delta_\alpha(k') \ = \ \lim_{k\to\infty} \phi_\alpha(x,k) \ = \ \nu(\alpha|x)$$

For $0.\omega \geq \sum_\alpha \nu(\alpha|x)$ no k is found, and T runs forever with no output (which is fine). ■

***O*-sampled conditionals.** Without O access, Theorem 6 shows that a semimeasure has l.s.c. conditionals iff it is sampled by a pTM. However, because the oracle O may be probabilistic, it is not clear that the $\Leftarrow$ direction still holds with oracle access. Therefore, we will avoid Leike et al.'s [LTF16] potentially misleading terminology "l.s.c. with oracle access" for these semimeasures. Instead, we will say that a semimeasure μ has O-sampled conditionals (or "is O-sampled" for brevity) if there is a pTM T such that $\mu(\alpha|x) = \lambda_T^O(\alpha|x)$ for $\alpha \in \mathcal{A}$.

***O*-estimable conditionals.** Following the convention set by "O-sampled" conditionals, we will use the term "O-estimable" conditionals to refer to semimeasures that have conditional probabilities estimable with O access. When it is clear from context we will drop the word "conditionals."

Formalizing oracles. For our purposes, oracles always answer queries with 0 or 1 (which can be interpreted as false or true). Because they are allowed to (independently) randomize their answers on queries, an oracle's behavior is specified by its probability of answering 1. This means we can treat oracles as functions to the unit interval.

Definition 7 (reflective oracle) An oracle $O : \mathcal{T} \times \mathcal{A}^* \times (\mathbb{Q} \cap [0,1]) \times \mathcal{A} \to [0,1]$ is called reflective iff for each pTM T and string $x \in \Sigma^*$, $\exists \{q_\alpha\}_{\alpha\in\mathcal{A}}$ satisfying the following properties:

$$\sum_{\alpha\in\mathcal{A}} q_\alpha = 1 \tag{2}$$

And for all $\alpha \in \mathcal{A}$ and $p \in \mathbb{Q}$,

$$\lambda_T^O(\alpha|x) \leq q_\alpha \leq 1 - \sum_{\beta\neq\alpha} \lambda_T^O(\beta|x)$$

$$O_\alpha(T,x,p) = 1 \quad \text{for} \quad p < q_\alpha$$
$$O_\alpha(T,x,p) = 0 \quad \text{for} \quad p > q_\alpha$$

This is the same as Definition 32 restated to take advantage of our λ_T^O notation. We will often abbreviate "reflective oracle" as rO.

We will reserve the notation $O_\alpha(T,x,p) \to 0$ (respectively 1) for the event that reflective oracle O_α called on the query (T,x,p) yields response 0 (respectively 1). This occurs with probability $O_\alpha(T,x,p)$ by definition, so "calling" $O_\alpha(T,x,p)$ is equivalent to invoking $\text{flip}(O_\alpha(T,x,p))$ where $\text{flip}(p)$ is a function that returns 1 with probability p and 0 with probability $1-p$.

Because of equation (2), q_α can be viewed as a conditional probability assignment for each symbol $\alpha \in \mathcal{A}$. When λ_T^O is a measure, the query (T,x,p) can be viewed as asking the question "is (case 0) $p > \lambda_T^O(\alpha|x)$ or (case 1) $p < \lambda_T^O(\alpha|x)$?" then O's answers are consistent with $q_\alpha = \lambda_T^O$; always 1 when $p < q_\alpha$ and 0 when $p > q_\alpha$, but allowed to randomize when $p = q_\alpha$ exactly. This randomization means that q_α cannot be determined exactly and avoids diagonalization. When λ_T^O is only a defective semimeasure, its conditionals do not sum to 1 so cannot satisfy equation (2), which means that $q_\alpha \neq \lambda_T^O(\alpha|x)$; however the definition requires at least $q_\alpha \geq \lambda_T^O(\alpha|x)$. This means that O "redistributes" the non-halting probability mass of T^O and completes λ_T^O to a measure. The requirement $q_\alpha \leq 1 - \sum_{\beta\neq\alpha} \lambda_T^O(\beta|x)$ is actually redundant because it follows from $q_\alpha \geq \lambda_T^O(\alpha|x)$ and $\sum_{\alpha\in\mathcal{A}} q_\alpha = 1$. The existence of reflective oracles on non-binary alphabets is proven in Appendix B.

Fallenstein et. al. originally defined reflective oracles for a binary alphabet in an analogous way [FTC15]. Leike [LTF16] used a more general definition which allowed O to randomize arbitrarily in the entire range $\lambda_T^O(\alpha|x)$ to $1 - \sum_{\beta\neq\alpha} \lambda_T^O(\beta|x)$[4]. When it is necessary to distinguish between the two cases we will call reflective oracles satisfying Fallenstein et. al.'s and our stricter definition "step reflective oracles."

Let $\bar{\lambda}_T^O$ be the completion of λ_T^O by a reflective oracle O, with $\bar{\lambda}_T^O(\alpha|x) = q^O_{\alpha,T,x}$, where $q^O_{\alpha,T,x} = q_\alpha$ as defined in Definition 7. This is a properly normalized probability measure by equation (2). Note that $\bar{\lambda}$ is a function of O and T (producing a measure). The completion (bar) is not applied as an operator to λ_T^O, because many different pTM's may produce the same semimeasure which can be completed in different ways. For example when T does not make oracle calls and $\lambda_T^O = \lambda_T$ is defective (say, 0 everywhere) it can be completed arbitrarily with appropriate choice of rO as mentioned in Appendix B.

Theorem 8 (properties of $\bar{\lambda}_T^O$) For any pTM T, $\bar{\lambda}_T^O$ is an O-estimable probability measure. In particular, there is an oracle pTM B_T estimating $\bar{\lambda}_T^O$ that is computably constructable from T.

[4]This definition is only specified for the binary case and is not easy to directly extend to the non-binary case. It is possible that Leike adopted the more general definition to simplify his proof of limit-computability.

Proof. Given any reflective oracle O, for each pTM T, and string x there are particular (clearly unique) values $q^O_{\alpha,T,x}$ satisfying the above requirements for q_α. There is a pTM B_T with O access that conducts a binary search for $q^O_{\alpha,T,x}$ by using queries to O to determine whether each p is above or below $q^O_{\alpha,T,x}$. This process may behave stochastically if $q^O_{\alpha,T,x}$ itself is ever a query, but the limit is always correct. Since the range of possible values for $q^O_{\alpha,T,x}$ halves with each query, it is O-estimable. ■

Notably, our procedure for estimating $\bar{\lambda}^O_T$ does not involve simulating T as in the procedure to l.s.c. λ_T (which does not work with oracle access), but only uses the description of T to run the binary search B_T. This is related to λ^O_T *only* because reflectivity of O leads to $\bar{\lambda}^O_T \geq \lambda^O_T$.

Reflective Oracles and Diagonalization. Let $T \in \mathcal{T}$ be a probabilistic Turing machine with a two symbol output alphabet $\mathcal{A} = \{\alpha, \beta\}$ that outputs β if $O_\alpha(T, \epsilon, 1/2) \to 1$ and α if $O_\alpha(T, \epsilon, 1/2) \to 0$. ($T$ can know its own source code by quining [Kle52, Thm. 27]). In other words, T queries the oracle about whether it is more likely to output α or not, and then does whichever the oracle says is less likely. In this case we can use an oracle $O_\alpha(T, \epsilon, 1/2) := 1/2$ (answer 0 or 1 with equal probability), which implies $\lambda^O_T(\alpha|\epsilon) = \lambda^O_T(\beta|\epsilon) = 1/2$, so the conditions of Definition 7 are satisfied. In fact, for this machine T we must have $O_\alpha(T, \epsilon, 1/2) = 1/2$ for all reflective oracles O.

Theorem 9 (pTM for $\bar{\lambda}^O_T$) For any reflective oracle O, all O-estimable semimeasures are O-sampled. In particular, for any pTM T, $\bar{\lambda}^O_T$ is O-sampled.

Proof. Any estimable function is also l.s.c. since the lower bound of the estimate can be used as the approximation from below. This means that the sampling algorithm (Algorithm 7) can be used to sample from any O-estimable probability measure (in this case halting with probability 1). Therefore, $\bar{\lambda}^O_T$ is also O-sampled; assuming that pTM S implements the sampling algorithm and accepts its argument ϕ_α in the form of a pTM encoding, and that the binary search pTM B_T returns the low end of its interval estimates, $\bar{\lambda}^O_T = \lambda^O_{S(B_T,\cdot)}$. ■

More succinctly, "O-estimable conditionals" implies "O-sampled conditionals". The converse does not hold because a semimeasure is not necessarily equal to its completion, but the converse does hold for probability measures. See Appendix D for proofs using Leike et al.'s definition.

Lemma 10 (all estimable measures O-sampled) For any (joint) estimable measure ν there exists one pTM T such that $\nu = \bar{\lambda}^O_T$ regardless of the choice of reflective oracle O.

Proof. This theorem is a stronger version of Theorem 9 that applies when ν is estimable without oracle access, but requires only $\nu(x)$ to be estimable, not the conditional $\nu(\cdot|x)$. Let T sample its output α from an estimate of $\nu(x\alpha)/\nu(x)$. This is

estimable except when $\nu(x) = 0$, so for $\nu(x) \neq 0$, $\lambda_T^O(\alpha|x) = \nu(\alpha|x)$. When $\nu(x) = 0$, T may never halt and O completes $\lambda_T^O(\cdot|x)$ in some arbitrary way. This only affects conditionals for strings that already have probability 0 so the product defining $\bar{\lambda}_T^O$ still assigns all continuations probability 0 and $\bar{\lambda}_T^O = \nu$. ■

The original construction of O in [FTC15] involved a non-constructive fixed-point argument implicitly invoking a continuous "hierarchy" of oracles. It looked like that $\bar{\lambda}_T^O$ may not even be expressible within the arithmetic hierarchy. Surprisingly, we can choose O so that $\bar{\lambda}_T^O$ is limit-computable (without requiring O access, instead limit-computing O) by the following result:

Theorem 11 (a limit-computable reflective oracle [LTF16, Thm.6])
There is a limit-computable (binary alphabet) reflective oracle.

We show in Theorem 43 that there are also limit-computable non-binary alphabet reflective oracles.

5 Multi-Player Games

Now we are ready to formally define multi-player games and strategies. We will show how multi-player games give rise to a subjective environment for each player. We refer to Bayes-optimal strategies in a subjective environment as Bayesian strategies in the associated multi-player game. Next we use the computability results established in Section 4 to introduce reflective-oracle computable strategies and show they are effectively enumerable, which prepares us to describe players' beliefs with Bayesian mixture strategies. Together these results allow us to describe Bayesian players who believe that strategies are rO-computable.

5.1 Definitions

We define multi-player games following [LTF16, Sec.7.3]: In a multi-player game, n players take sequential actions from $\mathcal{A}$ independently and in parallel. In step t, the game receives a vector of actions $a_t \in \mathcal{A}^n$ where action $a_t^i \in \mathcal{A}$ corresponds to player i. The history of actions including a_t determines a stochastic "move by nature" containing an n percept vector $e_t \in \mathcal{E}^n$ where player i only sees $e_t^i \in \mathcal{E}$. Players can only see their own actions (though of course the percept might include the other players' actions in some games). As before, $e_t^i = o_t^i r_t^i$ where $r_t^i \in [0, 1]$ is a reward.

Formally,

Definition 12 (multi-player game) A multi-player game is a function

$$\sigma : (\mathcal{A}^n \times \mathcal{E}^n)^* \times \mathcal{A}^n \to \Delta(\mathcal{E}^n)$$

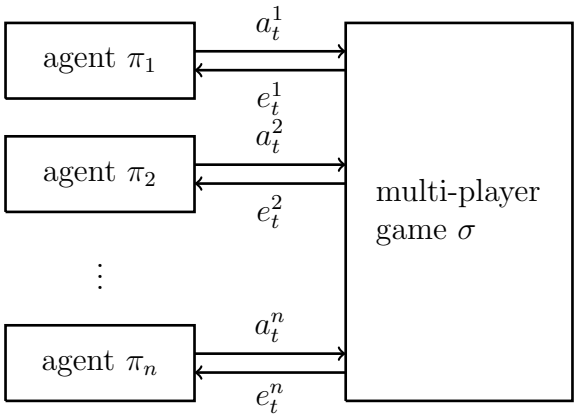


Figure 1: Agents $\pi_1, \ldots, \pi_n$ interacting in a multi-player game.

The interaction of the player strategies $\pi = (\pi_1, \pi_2, ..., \pi_n)$ with the multi-player game σ induces a history distribution $\sigma^\pi = \sigma^{\pi_{1:n}}$ where

$$\sigma^\pi(\varepsilon) \;:=\; 1$$

$$\sigma^\pi(æ_{1:t}) \;:=\; \sigma^\pi(æ_{<t}a_t)\sigma(e_t|æ_{<t}a_t)$$

$$\sigma^\pi(æ_{<t}a_t) \;:=\; \sigma^\pi(æ_{<t})\prod_{i=1}^{n}\pi_i(a_t^i|æ_{<t}^i)$$

Because players choose their actions simultaneously, the action distributions at time t are independent conditional on the action observations history $æ_{<t}^i := a_1^i e_1^i a_2^i e_2^i ... a_{t-1}^i e_{t-1}^i$, so we take a product. The history distribution for player i is the history distribution σ^π marginalized over the actions and observations of the other players:

$$\sigma_i^\pi(æ_{<t}^i) \;:=\; \sum_{æ_{<t}^j, j\neq i} \sigma^{\pi_{1:n}}(æ_{<t})$$

The subjective environment $\sigma_i(e_t^i|æ_{<t}^i a_t^i) = \sigma_i^\pi(e_t^i|æ_{<t}^i a_t^i)$ for single player/agent i is actually independent of π_i (see Appendix F), though it does depend on π_j for $j \neq i$. Therefore we will sometimes use σ_i^π to refer to the subjective environment. In the single agent-environment setting [Hut05] σ_i corresponds to the true environment which we write as $\mu \in \mathcal{M}$ and no superscripts i.

Definition 13 (environment) An environment μ is a chronological action-contextual measure. Equivalently, μ can be specified by its conditional probabilities $\mu(\cdot|h_{<t}a_t) \in \Delta\mathcal{E}$ given every history $h_{<t} = a_1 e_1 ... a_{t-1} e_{t-1}$ that it assigns a nonzero probability.

5.2 Strategies

Definition 14 (reflective-oracle computable strategies) Given a reflective oracle O for action space $\mathcal{A}$ and pTM's with alphabet $\Sigma = \mathcal{A} \bigsqcup \mathcal{E}$[5], so that $O's$ second argument is in Σ^* and O is indexed by $\mathcal{A}$, we say that strategy π is reflective-oracle computable (equivalently O-sampled or O-estimable) iff for some oracle pTM T, $\forall a_{<t} \in \mathcal{A}^{t-1}$ and $e_{<t} \in \mathcal{E}^{t-1}$ and $a \in \mathcal{A}$ we have $\pi(a|æ_{<t}) = \lambda_T^O(a|æ_{<t})$. We will abbreviate reflective-oracle computable as "rO-computable" and refer to this class of strategies as $\mathcal{P}^O_{\text{refl}}$.

Enumerability of $\mathcal{P}^O_{\text{refl}}$. Note that equivalence holds by Theorem 8 and Theorem 9 because all strategies are assumed to be (chronological, observation contextual, proper) probability measures, because all Section 4 theorems immediately generalize to allow input strings over Σ. The class of rO-computable strategies is effectively enumerable as $\bar{\lambda}_T^O$ for $T \in (T_1, T_2, ...)$ an effective enumeration of pTMs. This enumeration contains all rO-computable measures because oracle completion leaves probability measures unchanged. Conversely, all $\bar{\lambda}_T^O$ are O-sampled by Theorem 9, which means they are rO-computable.

6 Reflective-Oracle Computable Nash Equilibrium

We want to construct a set of mutually expected reward maximizing strategies $\pi^*_{\sigma_1}, \pi^*_{\sigma_2}, ..., \pi^*_{\sigma_n}$ for σ. It is not obvious that this is possible because the optimal strategy for each player depends on every other players' strategy. Explicitly, $\pi^*_{\sigma_i}$ depends on σ_i which depends on each other $\pi^*_{\sigma_j}$, which itself depends on σ_j, which depends (circularly) on $\pi^*_{\sigma_i}$. However, given an assignment of strategies to players it is certainly well-defined to discuss whether each of them is optimal given the others (i.e. a best response).

Theorem 15 (subjective environment estimable) Given pTM's generating the multi-player game σ and oracle pTM's generating the strategies $\pi_1, \pi_2, ..., \pi_n \in \mathcal{P}^O_{\text{refl}}$ there is an algorithm that constructs oracle pTM's estimating σ^π, σ^π_i, and σ_i.

Proof. Because σ is sampled by a pTM, it is l.s.c. by Theorem 6. Because it is a l.s.c. probability measure it is estimable. Because σ^π, σ^π_i, and σ_i are defined by uniformly continuous operations on σ and $\pi_1, ..., \pi_n$, their conditionals are also O-estimable (by computably constructable oracle pTM's). ■

[5]It is acceptable for pTM's to output any symbol in Σ as long as O is indexed by $\mathcal{A}$ so that conditionals are completed to $\Delta\mathcal{A}$. Producing the wrong type of output is treated the same as failing to halt. Later we will explicitly introduce types for symbols, allowing O to be indexed by any symbol of Σ so that the conditionals for symbols in $\mathcal{E}$ are also completed to $\Delta\mathcal{E}$.

By an optimal strategy, we mean one that maximizes the expected sum of a player's discounted future rewards. Formally, a discount factor $\gamma_i \geq 0$ scales the reward at step i to ensure the sum is finite. We will assume w.l.o.g. that the rewards are bounded so that $\sum_{i=1}^{\infty} \gamma_k < \infty$ ensures that $\sum_{i=t}^{\infty} \gamma_i r_i$ always exists. We define the discount normalization factor $\Gamma_t = \sum_{i=t}^{\infty} \gamma_i$, and

Definition 16 (value function) The value function of strategy π interacting with (subjective) environment ν is

$$V_\nu^\pi(h_{<t}) \;=\; \frac{1}{\Gamma_t} \lim_{T\to\infty} \sum_{æ_{t:T}} \sum_{i=t}^{T} \gamma_i r_i \prod_{j=t}^{T} \pi(a_j|h_{<t}æ_{<j})\nu(e_j|h_{<t}æ_{<j}a_j)$$

which satisfies the Bellman equation

$$V_\nu^\pi(h_{<t}) \;=\; \frac{1}{\Gamma_t} \sum_{a_t,e_t} \pi(a_t|h_{<t})\nu(e_t|h_{<t}a_t)(\gamma_t r_t + V_\nu^\pi(h_{<t}æ_t))$$

In a multiplayer game σ, player i's value function in his subjective environment $V_{\sigma_i}^{\pi_i}$ is in game theoretic terms [KL93b] his expected utility $U_i(\pi)$.

Definition 17 (optimal strategy π_ν^*) An optimal strategy π_ν^* for environment ν is a strategy in $\operatorname{argmax}_\pi V_\nu^\pi$, which is nonempty by [LH14]. Note that maximizing V_ν^π for different histories $h_{<t}$ is not in conflict, as can be seen from the Bellman equation. We define $V_\pi^* = V_\nu^{\pi_\nu^*}$ (which does not depend on the choice of π_ν^*). Clearly $\pi_\nu^*(\cdot|h_{<t})$ must be supported on $\operatorname{argmax}_a V_\nu^*(h_{<t}a)$, where V_ν^* is extended naturally to histories ending with an action. Because Γ_t is a positive scale factor, maximizing the value function is equivalent to maximizing the expected sum of discounted future rewards.

To prove the existence of a rO-computable Nash equilibrium, we need one more result which is of independent interest.

Theorem 18 (oracle-computable optimal strategy) For any environment ν whose conditionals are O-estimable, and any estimable discount normalization factor Γ_t, there is a rO-computable optimal strategy π_ν^*.

Proof. Note that environments do not produce elements of $\mathcal{A}$ so cannot be completed with O; this means that O-estimability is stronger than O-sampled conditionals (even though they are probability measures). Later we will introduce typed reflective oracles and define reflective-oracle computability for environments; adopting such an oracle closes this gap.

The optimal value function V_ν^* for ν with discount factor γ_k and discount normalization factor $\Gamma_t = \sum_{i=t}^{\infty} \gamma_i$, is

$$V_\nu^*(h_{<t}a_t) \;=\; \frac{1}{\Gamma_t} \lim_{T\to\infty} \sum_{e_t} \max_{a_{t+1}} \sum_{e_{t+1}} \ldots \max_{a_T} \sum_{e_T} \sum_{i=t}^{T} \gamma_i r_i \prod_{j=t}^{T} \nu(e_j|h_{<t}æ_{<j}a_j) \qquad (3)$$

We assume that both γ_t and Γ_t are estimable. This is true in the most common cases that $\gamma_t = \gamma^t$ for a constant rational $\gamma \in (0,1)$ or $\gamma = 1$ until some finite horizon after which it is 0. Our assumption is stronger than estimability of γ_t which would only make Γ_t l.s.c. but estimability of Γ_t (for all t) immediately implies estimability of γ_t.

Now all quantities in the limit of equation (3) are estimable. The limit can be approximated from below by iteratively increasing T. Recalling that the rewards are bounded to [0,1], this approximation is also within Γ_{T+1} of an upper bound because this is the maximum possible return for the rounds after T. This means that the limit can be approximated both from above and below hence is estimable. The factor $1/\Gamma_t$ is also estimable when $\Gamma_t > 0$, but when $\Gamma_t = 0$ the unnormalized value function is also 0 and we will not need to estimate it (any action is equally good).

It would be natural to guess that since the values for each action are O-estimable, one can simply compute them to sufficient precision and choose the best. However, this does not deal with ties between action values. Instead we need to take advantage of O access again. Noting that the value function is in $[0,1]$ we can use Theorem 9 to construct a TM $T_{\alpha\beta}$ such that

$$\begin{aligned}
\lambda^O_{T_{\alpha\beta}}(\alpha|æ_{<t}) &= \tfrac{1}{2}[V^*_\nu(æ_{<t}\alpha) - V^*_\nu(æ_{<t}\beta) + 1] \in [0;1] \\
\lambda^O_{T_{\alpha\beta}}(\beta|æ_{<t}) = 1 - \lambda^O_{T_{\alpha\beta}}(\alpha|æ_{<t}) &= \tfrac{1}{2}[V^*_\nu(æ_{<t}\beta) - V^*_\nu(æ_{<t}\alpha) + 1] \in [0;1]
\end{aligned}$$

where α and β are actions. Then in the two action case we define

$$\pi(a|æ_{<t}) = \begin{cases} 1 \text{ if } a = \alpha \text{ and } O(T_{\alpha\beta}, æ_{<t}, 1/2) \to 1, \\ 1 \text{ if } a = \beta \text{ and } O(T_{\alpha\beta}, æ_{<t}, 1/2) \to 0, \\ 0 \text{ otherwise.} \end{cases}$$

The procedure described above simply calls O once and chooses an action based on the response. Recall the notation $O(T,x,p) \to 0$ or $O(T,x,p) \to 1$ indicates that an oracle call with query (T,x,p) yields 0 or 1 (respectively). Since the oracle's behavior is stochastic this does not necessarily mean that $O(T,x,p)$ is valued at 0 or 1.

When $V^*_\nu(æ_{<t}\alpha) > V^*_\nu(æ_{<t}\beta)$, π takes action α, and when $V^*_\nu(æ_{<t}\alpha) < V^*_\nu(æ_{<t}\beta)$, π takes action β. When the action values are exactly equal, then π randomizes in a fashion depending on O, but in this case any action choice is equally good. Because π optimizes the optimal value function it is an optimal strategy.

If the action set is larger than 2, as suggested by [FTC15], we can construct a version of $T_{\alpha\beta}$ for each pair of actions, then use O to iteratively compare each action not yet tested against the best so far to find one with the maximum action value. ■

Theorem 19 (Nash equilibrium) For any multi-player game σ with l.s.c. conditionals, mutually optimal response strategies $\pi^*_1, ..., \pi^*_n$ exist and are reflective-oracle computable.

Proof. By Theorem 15, there is an algorithm to construct σ_i from oracle pTM's for $\sigma, \pi_1, ..., \pi_n$. There is also an algorithm to construct $\pi^*_{\sigma_i}$ from σ_i following the proof

of Theorem 18. Combining these two algorithms we obtain an algorithm T_i that constructs $\pi^*_{\sigma_i}$ from $(T_\sigma, T_{\pi_1}, ..., T_{\pi_n})$, following once more the convention that T_μ is an oracle pTM that samples μ. We now have to show that there are π_i such that the constructed optimal responses $\pi^*_{\sigma_i}$ w.r.t. environments σ_i^π give back π_i, i.e. that $\exists \pi_i : \pi^*_{\sigma_i} = \pi_i$. Define T'_i to run the oracle pTM returned by $T_i(T_\sigma, T'_1, ..., T'_n)$. This relies on the second recursion theorem implicitly: Let pTM A accept a two input pTM T and an input y and construct a new TM $T_y(x) = T(x, y)$. There is a machine $T'(x, i)$ that obtains its own description and runs $N(x)$ where $N = T_i(T_\sigma, A(T', 1), ..., A(T', n))$. Formally $T'_i = A(T', i)$. Every step in the process of running T'_i has already been shown to halt, so it samples from the optimal strategy $\pi^*_{\sigma_i}$ (meaning that $N = T_{\pi^*_{\sigma_i}}$). ■

Each strategy is optimal given the knowledge of all other players' strategies. Players even act optimally on the histories that they play with probability zero, so this is a subgame perfect Nash equilibrium.

7 Convergence for Bayesian Players

We have shown the existence of a reflective-oracle computable Nash equilibrium, which concerns the case that all players know each other's strategies. It is more interesting to consider Bayesian players that do not know each other's strategies, but only have some belief distribution over possible strategies they may face. It is typically difficult (or impossible) to show convergence for Bayesian players in general environment classes or games; see for example [LH15]. The main obstacle is that players may believe exploration is too dangerous. Kalai and Lehrer [KL93a] showed that in an infinitely repeated game with perfect monitoring Bayesian players can learn to play an approximate Nash equilibrium, supporting the centrality of Nash equilibria to game theory[6]. This is a particularly impressive result because it shows convergence for purely rational players (without requiring artificial exploration as in e.g. Thompson sampling) to a randomized strategy, despite the fact that Bayes optimal strategies can always be made deterministic[7]. Any solution to the grain of truth problem gets around this apparent contradiction because the deterministic Bayes optimal strategies may not appear in $\mathcal{P}$. The catch is that, informally, each player must assign a small positive probability (a "grain of truth") to the strategies actually chosen by his opponents. Well known impossibility theorems ([Nac97; FY01]) have suggested that this condition is hard or impossible to meet and limited the applicability of Kalai and Lehrer's results. Indeed, it took 22 years for the first non-trivial such class to be found [FST15; FTC15; LTF16]. We will show how reflective oracles can be used to construct a grain of truth by taking advantage of the effective enumeration of $\mathcal{P}^O_{\text{refl}}$ to find a dominant "mixture" strategy ζ. It is then straightforward to construct

[6] Or depending on one's perspective, justifying the Bayesian approach to game theory.

[7] See [FY01] for an explanation of further difficulties

Bayesian players whose beliefs are consistent with any strategy in the rich class $\mathcal{P}^O_{\text{refl}}$ that satisfy the conditions of Kalai and Lehrer's result. Our novel[8] result shows that Nash equilibria arise very naturally in infinitely repeated stage games, at least insofar as it is natural to supply players with a common reflective oracle.

Infinitely repeated games of Kalai and Lehrer. Kalai and Lehrer require that each player i maintains independent belief distributions over the strategy of all players. Player i's uncertainty about which strategy in $\mathcal{P}$ player j has chosen can be expressed as mixture of behavior strategies in $\mathcal{P}$, and is itself a behavior strategy by Kuhn's theorem [Aum64] (though in general it may not be in $\mathcal{P}$). Therefore we can write it as π^i_j, with superscript representing the player who's state of knowledge we are considering and the subscript representing the player he is reasoning about, so that player i's full beliefs about the strategies of all players is given by a vector $\pi^i = (\pi^i_1, \pi^i_2, ..., \pi^i_n)$. This also allows us to represent more general beliefs that might not be constructed as a Bayesian mixture over a strategy class. Every player at least knows his own strategy so $\pi^i_i = \pi_i$. The true strategy vector is given by $\pi_{1:n} = (\pi_1, \pi_2, ..., \pi_n)$ as in Section 6 (we will sometimes suppress the subscripts $1:n$ in $\pi_{1:n}$). The lack of a superscript indicates that this is not subjective. We assume that the reward for player i is specified by a fixed payoff function u_i depending only on the actions of all players in the current round. Each player knows his own payoff function (since the action sets are finite, u_i is sometimes called a payoff matrix, and for our purposes may be assumed computable without any significant loss in generality). Though player i does not know any other player's payoff function, that information would not be useful anyway because he does not assume other players' policies to be optimal. Perfect monitoring means that each player observes the other players' actions; there are no further observations. Therefore σ is a multi-player environment as defined above but with additional restrictions; in particular there is no longer a meaningful difference between σ^π and σ^π_i, because $e^i_t = a^{\neq i}_t = a^1_t ... a^{i-1}_t a^{i+1}_t ... a^n_t$. This means that the history distribution σ^π contains multiple copies of the same action history as distributed to each player through σ^π_i. It is now the case that a player's beliefs about his subjective environment depend on both his index in σ and his beliefs about the strategies of other players, so that player i models his environment as $\sigma_i^{\pi^i}$ which is not in general the same as his subjective environment σ^π_i.

Conditions for convergence. Kalai and Lehrer's result requires that π_i acts rationally with respect to player i's beliefs, or in our terminology that $\pi_i = \pi^*_{\sigma_i^{\pi^i}}$. This is not a circular definition because $\sigma_i^{\pi^i}$ does not depend on π^i_i (when viewed as an environment), see Appendix F. Additionally, they require that $\sigma^\pi \ll \sigma^{\pi^i}$, which follows from the grain of truth property.

[8]Though [LTF16] suggests that Kalai and Lehrer's conditions can be satisfied with reflective oracles, they do not provide a proof or even explicitly construct the appropriate strategy class $\mathcal{P}^O_{\text{refl}}$. Also, they claim that all players must know the others are Bayesian, which is not required.

Constructing a mixture policy. We want to satisfy the convergence conditions of Kalai and Lehrer, but this could be done without learning by setting each player's beliefs π^i to the true optimal strategies $\pi^* = (\pi^*_{\sigma_1}, ..., \pi^*_{\sigma_n})$ as in Section 6; for our purposes the players must also hold (independent) priors distributed over all of $\mathcal{P}^O_{\text{refl}}$ to model their ignorance of each opponent's strategy. We will actually satisfy a slightly different condition by finding a *dominant* strategy $\zeta \in \mathcal{P}^O_{\text{refl}}$ such that $\forall \pi \in \mathcal{P}^O_{\text{refl}}$, $\exists c \in \mathbb{R}^+$ such that $\zeta(\cdot) \geq c\pi(\cdot)$. Our usage of the term "dominant strategy" is not related to the usual game-theoretic meaning; it is a measure-theoretic property not an optimality property. A Bayesian mixture $\sum_{\pi \in \mathcal{P}^O_{\text{refl}}} w_\pi \pi(\cdot)$ satisfies the latter with $c = w_\pi > 0$. Bayesian mixtures over $\mathcal{P}$ are not always in $\mathcal{P}$ but we show below that this holds if $\sum_\pi w_\pi = 1$, so that we could simply define ζ this way and it would be a probability measure and therefore a strategy (so the following algorithm is unnecessary). However, the simplicity based priors often used in algorithmic information theory [LV+08], including to define Solomonoff induction and AIXI [HQC24] are only l.s.c. semimeasures. The following construction for ζ encompasses the general case that the weights may be only l.s.c., which only happens when they are defective ($\sum w_\pi < 1$) because l.s.c. probability measures are estimable. A player with prior ζ still learns any opponent's strategy in $\mathcal{P}^O_{\text{refl}}$ in the sense of strong merging [KL93a], so ζ models an unknown strategy, and it can also be used to satisfy Kalai and Lehrer's conditions when all players are Bayesian. Fix l.s.c. weights $w_\pi > 0$ for each $\pi \in \mathcal{P}^O_{\text{refl}}$. For any pTM T let π_T be the strategy corresponding to the measure $\bar{\lambda}^O_T$, and consider TM Q implementing Algorithm 1.

Algorithm idea. We would like to sample from $\zeta' = \sum_\pi w_\pi \pi$, but because we want loose requirements on the computability of w_π we cannot assume they sum to 1. This means we would like to complete ζ'. Unfortunately we cannot do this either because though ζ' is O-l.s.c., its conditionals $\zeta'(a_{\leq t}||e_{\leq t})/\zeta'(a_{<t}||e_{<t})$ involve division by an O-l.s.c. quantity so may not be O-l.s.c. themselves; in particular we do not even know if ζ' is O-sampled[9] so we cannot produce an oracle pTM T sampling it and there is no λ^O_T to complete. Instead, we define a new pTM Q to lower semicompute the numerator of the conditional, $\zeta'(a_{\leq t}||e_{\leq t})$, and use the completed $\bar{\lambda}^O_Q$ to estimate the denominator $\pi_Q(a_{<t}||e_{<t})$. This makes the fraction $\zeta'(a_{\leq t}||e_{\leq t})/\pi_Q(a_{<t}||e_{<t})$ O-l.s.c., which means we can sample from it. Then completing λ^O_Q we obtain our measure π_Q which may not literally complete ζ', but does dominate it. The core of this construction relies (again) on Kleene's second recursion theorem, in this case allowing Q access to its own description, which it needs to estimate the denominator, intuitively "pretending that it has already been completed."[10] Note that this is the only recursion within Q; we estimate $\bar{\lambda}^O_Q$ by running the binary search pTM B_Q, which makes oracle calls

[9]When w_π is estimable, the conditionals are estimable and therefore O-sampled; in [FST15] this is elegantly demonstrated by rejection sampling, though it is obvious in light of our Theorem 9. Their proof does not extend to l.s.c. w_π.

[10]This iterative completion is reminiscent of Solomonoff normalization [HQC24, Sec.2.8.2], but may not preserve the ratios of conditionals.

about Q but never actually simulates Q.

Algorithm 1 pTM Q

Input: History $æ_{<t}$
Require: Random sequence ω
Output: $a_t \sim \lambda_Q^O(a_t|æ_{<t})$
1: Obtain $\langle Q \rangle$
2: Let $\phi_\alpha(æ_{<t}, \cdot)$ approximate $\sum_{\pi \in \mathcal{P}^O_{\text{refl}}} w_\pi \frac{\pi(a_{<t}||e_{<t})}{\pi_Q(a_{<t}||e_{<t})} \pi(\alpha|æ_{<t})$ from below, where $\pi_Q \equiv \bar{\lambda}_Q^O$
3: Run sample(ϕ_α, $æ_{<t}$) with access to ω (Algorithm 7).

Algorithm correctness. Line 1 is possible by Kleene's second recursion theorem. Line 1 is doing most of the work; we need to show that the right hand side is O-l.s.c. Because w_π is assumed l.s.c. and π and π_Q are O-estimable by Theorem 8, every term of the sum is O-l.s.c. This means that the sum is O-l.s.c. (computing the k^{th} partial sum for $\phi_\alpha(\cdot, k)$). We have to show inductively that the denominator is never 0, but this is easy because there is a computable measure assigning any finite string nonzero probability. Therefore $\zeta \in \mathcal{P}^O_{\text{refl}}$.

By the correctness of the sampling algorithm,

$$\lambda_Q^O(a_t|æ_{<t}) \;=\; \sum_{\pi \in \mathcal{P}^O_{\text{refl}}} w_\pi \frac{\pi(a_{<t}||e_{<t})}{\pi_Q(a_{<t}||e_{<t})} \pi(a_t|æ_{<t})$$

Now we can choose our dominant policy as π_Q:

$$\zeta \;:=\; \pi_Q \;=\; \bar{\lambda}_Q^O \;\geq\; \lambda_Q^O$$

By definition $\zeta \in \mathcal{P}^O_{\text{refl}}$. It only remains to show that ζ dominates the class.

$$\begin{aligned}
\zeta(a_{\leq t}||e_{\leq t}) \;&=\; \zeta(a_{<t}||e_{<t})\zeta(a_t|æ_{<t}) \;\geq\; \zeta(a_{<t}||e_{<t})\lambda_Q^O(a_t|æ_{<t}) \\
&=\; \zeta(a_{<t}||e_{<t}) \sum_{\pi \in \mathcal{P}^O_{\text{refl}}} w_\pi \frac{\pi(a_{<t}||e_{<t})}{\pi_Q(a_{<t}||e_{<t})} \pi(a_t|æ_{<t}) \\
&=\; \sum_{\pi \in \mathcal{P}^O_{\text{refl}}} w_\pi \pi(a_{\leq t}||e_{\leq t}) \;\geq\; w_\pi \pi(a_{\leq t}||e_{\leq t}) \quad \forall \pi \in \mathcal{P}^O_{\text{refl}}
\end{aligned}$$

noting for the inequality that the O-completed measure for any machine is lower bounded by its semi-measure. As desired, $\zeta \in \mathcal{P}^O_{\text{refl}}$, and $\zeta \geq \mathcal{P}^O_{\text{refl}}$. This observation is sometimes written as

Theorem 20 ($\mathcal{P}^O_{\text{refl}}$ contains a dominant element) There exists $\zeta \in \mathcal{P}^O_{\text{refl}}$ s.t. $\forall \pi \in \mathcal{P}^O_{\text{refl}}$, $\exists c > 0$ such that $\zeta(\cdot) \geq c\pi(\cdot)$ (ζ multiplicatively dominates π). The first condition is that ζ is in the class and the second that ζ dominates the class ($\zeta \geq \mathcal{P}^O_{\text{refl}}$). Because both are satisfied we say that P^O_{refl} has a dominant element, written $\mathcal{P}^O_{\text{refl}} \geq \mathcal{P}^O_{\text{refl}}$.

When the weights w sum to 1, $\zeta \in \mathcal{P}^O_{\text{refl}}$ is a Bayesian mixture over $\mathcal{P}^O_{\text{refl}}$, but we don't actually need this. We only need the dominance property, and ζ dominates the defective Bayesian mixture $\sum_{\pi \in \mathcal{P}^O_{\text{refl}}} w_\pi \pi$ and consequently any $\pi \in \mathcal{P}^O_{\text{refl}}$.

A grain of truth. Choosing $\pi^i_j = \zeta$ for $i \neq j$, $\sigma_i^{\pi^i}$ is defined as a product over O-sampled policies ζ (with deterministic computable rewards)[11]. By Theorem 15, $\sigma_i^{\pi^i}$ has O-estimable conditionals, and by Theorem 18 there is a reflective-oracle computable optimal strategy $\pi^*_{\sigma_i^{\pi^i}} \in \mathcal{P}^O_{\text{refl}}$. Therefore, the policy class $\mathcal{P}^O_{\text{refl}}$ and any multi-player game with l.s.c. conditionals form a solution to the grain of truth problem. The setting of Kalai and Lehrer (infinitely repeated games with computable rewards) is a special case: Because $\forall j \ \pi^*_{\sigma_j^{\pi^j}} \ll \zeta$, it is easy to see that when $\pi = (\pi^*_{\sigma_1^{\pi^1}}, \pi^*_{\sigma_2^{\pi^2}}, ..., \pi^*_{\sigma_n^{\pi^n}})$, and $\pi^i = (\zeta, ..., \pi^*_{\sigma_i^{\pi^i}}, ..., \zeta)$, $\sigma^\pi \ll \sigma^{\pi^i}$. Finally, we can appeal to [KL93a, Thm. 2] to conclude the following:

Theorem 21 (close to ε-Nash equilibrium) In a computable infinitely repeated game σ, if $\pi^i_j = \zeta$ for $i \neq j$ and $\pi^i_i = \pi_i = \pi^*_{\sigma_i^{\pi^i}}$ (so all players are Bayesian), then for every $\varepsilon > 0$, σ^π-a.s. there is a time t_ε such that for all $t \geq t_\varepsilon$, σ^π plays ε-like the history distribution of a ε-Nash equilibrium.

The term ε-Nash equilibrium means that every players' expected utility is within ε of the best achievable given knowledge of the other players' strategies. The term "plays ε-like" is defined in [KL93a, Def. 2], relying on [KL93a, Def. 1] of "ε-close" measures. It means that with high probability the conditionals of the history distributions are close; Kalai and Lehrer point out this is a kind of Provably Approximately Correct (PAC) guarantee. Statements holding after time t_ε refer to measures conditioned on the history up to time t_ε.

Matching pennies example. In the game of *matching pennies* there are two agents ($n = 2$), and two actions $\mathcal{A} = \{\alpha, \beta\}$ representing the two sides of a penny. In each time step agent 1 wins if the two actions are identical and agent 2 wins if the two actions are different. The payoff matrix is as follows.

	α	β
α	1,0	0,1
β	0,1	1,0

We use $\mathcal{E} = \{0, 1\}$ to be the set of rewards (observations are vacuous) and define the multi-agent environment σ to give reward 1 to agent 1 iff $a^1_t = a^2_t$ (0 otherwise) and reward 1 to agent 2 iff $a^1_t \neq a^2_t$ (0 otherwise).

According to our result, when the game is known Bayesian players with prior $\mathcal{P}^O_{\text{refl}}$ eventually converge to a Nash equilibrium of the repeated game. When discounting is geometric with discount factor γ close to 0, this means they will approximately play the (only) Nash equilibrium of the stage game, randomizing uniformly between actions α and β.

[11] It does not depend on π^i_i, making its definition a slight abuse of notation.

Differing action sets. When each player has a different action set, in order to use a consistent reflective oracle for every player, it is necessary to consider the encodings of actions. Leike et al.'s (implicit) approach [LTF16] was to choose complete, prefix-free, binary codes for each type of symbol, which introduces no serious difficulties but means that their algorithms should properly be specified on the bit level. We would like to use non-binary reflective oracles to take a more elegant approach. The naive idea of combining the action sets for each player $\mathcal{A} = \bigsqcup_{1 \leq i \leq n} \mathcal{A}_i$ does not immediately work because player i's strategies must be have their conditionals completed to probability measures in $\Delta\mathcal{A}_i \neq \Delta\mathcal{A}$. The solution is slightly harder than using n separate reflective oracles, because the n reflective oracles would have to consistently answer queries about each other. Fortunately it is possible to use a *typed* reflective oracle as described in Appendix B to map actions from each action set to their own simplex.

8 Impossibility Results

Why does our solution to Kalai and Lehrer's grain of truth problem not violate the impossibility results from the literature? Assume we are playing an infinitely repeated game where in the stage game no agent has a weakly dominant action and the pure action maxmin reward is strictly less then the minmax reward. The impossibility result of [Nac97; Nac05] state that there is no class of policies $\mathcal{P}$ such that the following are simultaneously satisfied.

- *Learnability.* Each agent learns to predict the other agent's actions.
- *Caution and Symmetry.* The set $\mathcal{P}$ is closed under simple policy modifications such as renaming actions.
- *Purity.* There is an $\varepsilon > 0$ such that for any stochastic policy $\pi \in \mathcal{P}$ there is a deterministic policy $\pi' \in \mathcal{P}$ such that if $\pi'(æ_{<t}) = a$, then $\pi(a|æ_{<t}) > \varepsilon$.
- *Consistency.* Each agent always has an ε-best response available in $\mathcal{P}$.

In order to converge to an ε-Nash equilibrium, each agent has to have an ε-best response available to them, so consistency is our target. Learnability is immediately satisfied for any environment in our class if we have a dominant prior [KL93a]. For $\mathcal{P}^O_{\text{refl}}$ caution and symmetry are also satisfied since this set is closed under any computable modifications to policies. However, our class $\mathcal{P}^O_{\text{refl}}$ avoids this impossibility result because it violates the purity condition: Let $T_1, T_2, \ldots$ be an enumeration of $\mathcal{T}$. With action space $\mathcal{A} = \{0, 1\}$, consider the policy π that maps history $æ^i_{<t}$ to the action $1 - \text{flip}(O_1(T_t, æ^i_{<t}, 1/2))$. If T_t is deterministic, then π will take a different action than T_t for any history of length $t - 1$. Therefore no deterministic reflective-oracle-computable policy can take an action that π assigns positive probability to in every time step.

[FY01] present a condition that makes convergence to a Nash equilibrium impossible: if the player's rewards are perturbed by a small real number drawn from some continuous density ν, then for ν-almost all realizations the players do not learn to predict each other and do not converge to a Nash equilibrium. For example, in a matching pennies game, rational agents randomize only if the (subjective) values of both actions are exactly equal. But this happens only with ν-probability zero, since ν is a density. Thus with ν-probability one the agents do not randomize. If the agents do not randomize, they either fail to learn to predict each other, or they are not acting rationally according to their beliefs: otherwise they would seize the opportunity to exploit the other player's deterministic action.

But this does not contradict our convergence result: the class $\mathcal{P}^O_{\text{refl}}$ is countable and each $\nu \in \mathcal{P}^O_{\text{refl}}$ has positive prior probability. Perturbation of rewards with arbitrary real numbers is not possible. Even more, the argument given by [FY01] cannot work in our setting: the Bayesian mixture π_Q mixes over λ_T for all probabilistic Turing machines T. For Turing machines T that sometimes do not halt, the oracle decides how to complete λ_T into a measure $\bar{\lambda}^O_T$. Thus the oracle has enough influence on the exact values in the Bayesian mixture that the values of two actions in matching pennies can be made exactly equal.

9 Asymptotic Optimality in Unknown Games

We now go further and show convergence to equilibrium even when the game is unknown and is not-repeated but one infinitely long game, as long as the players use *asymptotically optimal* strategies instead of Bayes-optimal strategies.

Definition 22 (asymptotic optimality) A policy π is asymptotically optimal in mean in environment class $\mathcal{M}$ iff $\forall \mu \in \mathcal{M}$, $\mathbb{E}^\pi_\mu[V^*_\mu(h_{<t}) - V^\pi_\mu(h_{<t})] \to 0$ as $t \to \infty$.

In fact, we can show convergence even when the players are not initially aware of each others' existence. To do this, we must extend the environment class to all rO-computable environments (which we define below similarly to $\mathcal{P}^O_{\text{refl}}$). Because we would still like players to be included in the environment we need the oracle to be usable for computing either strategies or multi-player games. Then the entire arrangement of a multi-player game with the other players embedded is rO-computable. The situation is similar to differing action sets; it would be possible to simply give environments access to a reflective oracle that provides completed action probabilities, but unfortunately this could not be used to complete the action-conditional semimeasures generated by pTM's producing perceptions, which ultimately means that the environment class would not be effectively enumerable. Instead we use a typed reflective oracle capable of completing both percept and action distributions. After defining our environment class, we show that Thompson sampling policies converge to a Nash equilibrium.

Definition 23 ($\mathcal{M}^O_{\text{refl}}$) Fix alphabet $\Sigma = \mathcal{A} \bigsqcup \mathcal{E}$. Let O be a typed reflective oracle with input and output alphabet Σ. Let $\mathcal{M}^O_{\text{refl}}$ be the set of environments $\nu_T = \bar{\lambda}^O_T$, called r$O$-computable.

Unlike the class of environments or games with estimable conditionals, $\mathcal{M}^O_{\text{refl}}$ is effectively enumerable because halting issues are resolved by oracle completion.

Theorem 24 (convergence to equilibrium) Let σ be a reflective-oracle computable multi-agent game and let $\pi_1, ..., \pi_n$ be rO-computable policies that are asymptotically optimal in mean in the class $\mathcal{M}^O_{\text{refl}}$. Then for all $\varepsilon > 0$ and all $i \in \{1, ..., n\}$ the $\sigma^{\pi_{1:n}}$-probability that the policy π_i is an ε-best response converges to 1 as $t \to \infty$.

Proof. This is [LTF16, Thm. 28]. Following the argument of Theorem 15, subjective environments are in $\mathcal{M}^O_{\text{refl}}$. Because each policy is asymptotically optimal in mean in its subjective environment, Theorem 24 follows from the observation that convergence in mean implies convergence in probability for bounded random variables. Therefore,

$$\sigma^\pi_i[V^*_{\sigma_i}(æ^i_{<t}) - V^{\pi_i}_{\sigma_i}(æ^i_{<t}) \geq \varepsilon] \to 0 \text{ as } \ t \to \infty$$

so the probability that π_i plays an ε-best response converges to 1 as $t \to \infty$. ■

Thompson sampling. Now we only need to find a set of asymptotically optimal rO-computable strategies. It is normally assumed that Bayesian agents solve the exploration/exploitation problem in a principled way, so it is somewhat surprising that they are not (even weakly) asymptotically optimal in too general environment classes [Ors10]. Thompson sampling [Tho33; Lei+16] is an asymptotically optimal variation of the Bayesian rational strategy modified to increase exploration. Let the effective horizon $H_t(\varepsilon)$ be the minimum number of steps in the future such that the discount normalization factor is less than a ε fraction of the current discount normalization factor; in the case $\varepsilon = 1/2$ this is the "half-life" of Γ_t. Formally $H_t(\varepsilon) = \min_k\{k|\Gamma_{t+k}/\Gamma_t \leq \varepsilon\}$. Then Thompson sampling is described by Algorithm 2 and denoted by π_{TS}. Naturally it is parameterized by a class of environments and a p(oste)rior weight function w. We will choose $\mathcal{M}^O_{\text{refl}}$ for the class of environments, which means players initially are not even aware that their opponents exist (or of their number). Then the first condition of the strong grain of truth property is satisfied:

Theorem 25 (Strong grain of truth property, first condition) If σ is an O-estimable game and $\pi_{\neq i} \in (\mathcal{P}^O_{\text{refl}})^{n-1}$ then $\sigma^\pi_i \in \mathcal{M}^O_{\text{refl}}$.

Proof. This follows from a slight generalization of Theorem 15 to include O-estimable games. ■

If more prior knowledge is required we can instead enumerate $\mathcal{G} \times \mathcal{P}^{n-1}$. Note that even in this case w is a *joint* p(oste)rior over $\mathcal{M} = \mathcal{G} \times \mathcal{P}^{n-1}$, i.e. can model any collusion between players (and even the game itself).

If we only assume that the weights are lower semicomputable semimeasures, there is a chance that sampling from them fails. This means that the infinite loop may get stuck after finitely many iterations. It seems that Thompson sampling is not rO-computable (because it is not a contextual probability measure) without stronger assumptions on w, for instance O-estimability. We can rephrase the Thompson sampling algorithm as shown in Algorithm 3 to explicitly show how to sample actions on each step (reflective-oracle computably) without persistent memory instead of abstractly describing the behavior between resampling environments. Equivalence is similar to Kuhn's theorem [Aum64].

Algorithm 2 Thompson sampling strategy π_{TS}

Input: Percept stream $e_{1:\infty}$
Output: $a_{1:\infty} \sim \pi_{TS}(\cdot||e_{1:\infty})$

1: **while** true **do**
2: sample $\rho \sim w(\cdot|æ_{<t})$
3: follow π^*_ρ for $H_t(\varepsilon_t)$ steps

Algorithm 3 Stepwise Thompson sampling strategy π_{TS}

Input: History $æ_{<t}$
Output: $\pi_{TS}(a_t|æ_{<t})\ \forall a_t \in \mathcal{A}$

1: $t_0 \leftarrow 0$; $i \leftarrow 0$
2: **while** $t_i \leq t$ **do** { $t_{i+1} \leftarrow t_i + H_{t_i}(\varepsilon_{t_i})$; $i \leftarrow i+1$ }
3: $t' \leftarrow t_{i-1}$
4: $\pi_{TS}(a_t|æ_{<t}) := \sum_{\rho \in \mathcal{M}^O_{\text{refl}}} w(\rho|æ_{<t'}) \frac{\pi^*_\rho(æ_{t':t}|æ_{<t'})}{\pi_{TS}(æ_{t':t}|æ_{<t'})} \pi^*_\rho(a_t|æ_{<t})$

It still remains to show that Algorithm 3 is rO-computable. By definition,

$$w(\rho|æ_{<t}) \;=\; w(\rho)\frac{\rho(æ_{<t})}{\xi(æ_{<t})} \quad \text{where} \quad \xi(æ_{<t}) \;:=\; \sum_{\rho \in \mathcal{M}^O_{\text{refl}}} w(\rho)\rho(æ_{<t})$$

When w is O-estimable, also every factor above (assuming that the environment class is general enough that all finite history prefixes are possible) and therefore the posterior weights are O-estimable. For any $\rho \in \mathcal{M}^O_{\text{refl}}$, Theorem 18 shows that the optimal policies π^*_ρ are all O-estimable. This makes Line 4 of Algorithm 3 possible with O access.

Theorem 26 (Thompson sampling computability) For estimable Γ_t and normalized estimable prior w over over $\mathcal{M}^O_{\text{refl}}$, π_{TS} over $\mathcal{M}^O_{\text{refl}}$ is rO-computable.

Together, Theorem 25 and Theorem 26 show that $\mathcal{P}^O_{\text{refl}}$ and the class of O-estimable games satisfy the Thompson sampling version of the strong grain of truth property.

Technically, computing the horizon $H_t(\epsilon_t)$ exactly would require finitely computable Γ_t and ϵ_t, but the convergence of Thompson sampling only depends on $\epsilon_t > 0$ and $\epsilon_t \to 0$. As long as Γ_{t+k}/Γ_t is computed to sufficient precision to ensure the ratio between resampling steps decreases to 0 this is equivalent to Thompson sampling with an acceptable choice of ϵ_t.

(Un)normalized weights w. Recall that all normalized l.s.c. w are also estimable. Given that we generally assume in this paper that the weights are at least l.s.c., Theorem 26 only relies on the weights summing to one; without this requirement, Thompson sampling would sometimes fail to sample an environment and its behavior is under-specified! Generalizing to l.s.c. weights, it is natural to try to use the reflective oracle to somehow complete Thompson sampling. We could try to complete π_{TS}'s environment mixture ξ. Unfortunately this would not explicitly complete the weights which Thompson sampling needs access to; π_{TS} requires not a dominant environment but explicit coefficients. The reflective oracle could be used to directly complete each weight from an oracle pTM generating it (in the sense of outputting 1 with probability $w(\rho|æ_{<t})$ and otherwise failing to halt) but it is unclear whether the individually completed weights would still sum to 1.[12]

Combined with Theorem 26, Theorem 11 tells us that we can choose O to make Thompson sampling limit-computable, which lets us improve Theorem 24.

Theorem 27 (limit-computable convergence to equilibrium [LTF16, Cor.20]) There are limit-computable strategies $\pi_1, ..., \pi_n$ such that for any computable multi-agent game σ and for all $\varepsilon > 0$ and all $i \in \{1, ..., n\}$ the $\sigma^{\pi_{1:n}}$-probability that the policy π_i is an ε-best response converges to 1 as $t \to \infty$.

Since all π_i converge to ε-best responses, this implies that $\pi_{1:n}$ is asymptotically ε-Nash. We can say more about the computability level of $\mathcal{M}^O_{\text{refl}}$:

Theorem 28 ($\boldsymbol{\Delta_1 \subset \mathcal{M}^O_{\text{refl}} \subset \Delta_2}$) The class $\mathcal{M}^O_{\text{refl}}$ contains all (joint) estimable (normalized) environments (sometimes called $\mathcal{M}^{\text{msr}}_{\text{est}}$) and is contained in the class of measures with limit-computable conditionals.

Proof. The claim $\mathcal{M}^{\text{msr}}_{\text{est}} \subset \mathcal{M}^O_{\text{refl}}$ follows immediately from Lemma 10, and is in fact strict by a simple diagonalization argument in [LTF16]. The claim that $\mathcal{M}^O_{\text{refl}} \subset \Delta_2$ follows from Theorem 11. It is easy to see that $\Delta_1 \subset \mathcal{P}^O_{\text{refl}} \subset \Delta_2$ also holds by the same argument. ■

10 An Application to Self-Prediction

In Section 9 we derived a convergence result for players who are not initially aware of the multi-player game they are playing, or even that other players are involved. We can take this ignorance even further by allowing our player to be uncertain of even his own strategy as he selects each individual move. This is the setting of the Self-AIXI agent proposed by [Cat+23]. Though the problem may appear esoteric at first, it is of interest to reinforcement learning (RL) researchers. Model-based RL algorithms

[12] Really, what we would like to do is divide ξ by $\sum_\rho w(\rho)$ to normalize directly, but this is not even l.s.c.

often execute an expensive decision tree search to plan their future actions. This search can be narrowed by (iteratively) distilling the resulting policy into a model that guides action selection. The Self-AIXI agent can be viewed as an extreme case of this approach, entirely replacing planning with an interaction between self-model and environment-model akin to model- and value-based policy search methods but more general and principled. Arguably, the Self-AIXI framework also describes human planning; though we may make plans for our future actions, we do not know which strategy we will ultimately follow. Formally, a Self-AIXI policy is defined[13] as

Definition 29 (Self-AIXI) Let ζ and ξ be dominant elements of policy class $\mathcal{P}$ and environment class $\mathcal{M}$.

$$\begin{aligned} \pi_S(h_{<t}) &\in \operatorname{argmax}_{a_t \in \mathcal{A}} V_\xi^\zeta(h_{<t}a_t) \\ V_\xi^\zeta(h_{<t}a_t) &:= \frac{1}{\Gamma_t} \lim_{m\to\infty} \sum_{a_{t+1:m}, e_{t:m}} \sum_{i=t}^{m} \gamma_i r_i \prod_{j=t}^{m} \xi(e_j|æ_{<j}a_j) \prod_{j=t+1}^{m} \zeta(a_j|æ_{<j}) \end{aligned}$$

The results of [Cat+23] suggest convergence of Self-AIXI to π_ξ^* (when ξ is a dominant element of the class $\mathcal{M}_{\text{lsc}}^{\text{semi}}$ of environments given by contextual chronological l.s.c. semimeasures, π_ξ^* is called AIXI), but there are gaps remaining[14]. One problem is that they rely on $\pi_S \in \mathcal{P}$ without constructing any policy class (interesting or otherwise) with this property. Reflective oracles provide a natural example.

Theorem 30 (Self-AIXI in computable environment) Let O be a reflective oracle with input alphabet $\Sigma = \mathcal{A} \bigsqcup \mathcal{E}$ and output alphabet $\mathcal{A}$. Let $\mathcal{P} = \mathcal{P}_{\text{refl}}^O$ and $\mathcal{M}$ be any environment class containing a dominant element ξ with estimable conditionals. Then there exists a dominant $\zeta \in \mathcal{P}_{\text{refl}}^O$, and there is a stochastic $\pi_S \in \mathcal{P}_{\text{refl}}^O$ for ζ, ξ.

Proof. The existence of such a ζ follows from Theorem 20. The argument that $\pi_S \in \mathcal{P}_{\text{refl}}^O$ follows Theorem 18; the conditionals of ξ are assumed estimable and the conditionals of ζ are O-estimable. ■

Note that since we do not have an effective enumeration of the class of environments with estimable conditionals, we cannot choose it as $\mathcal{M}$ in Theorem 30, making the result somewhat less interesting. We can solve this problem by sharing the reflective oracle between the strategies and the environments as in Section 9:

[13] We require strategies and environments to be (contextual, chronological) probability measures. This definition generalizes from explicit Bayesian mixtures to any dominant elements of each class. Also, our Definition 29 does not maximize equation (3) of [Cat+23], which asserts linearity of the action value function Q_ξ^ζ with the incorrect coefficients (failing to update on the latest action) and is probably not the intended definition as it is inconsistent with the rest of the paper. This definition of Q_ξ^ζ would make Self-AIXI a kind of one-step causal decision theorist instead of an evidential decision theorist.

[14] We do not address their requirement that π_S is "reasonable off-policy" which is a technical and slightly unnatural condition that has not been shown for any combination of strategy and environment classes.

Theorem 31 (Self-AIXI in oracle computable environment) Fix some finite alphabet Σ and encodings over Σ for each element of $\mathcal{A}$ and $\mathcal{E}$. Let O be a typed reflective oracle with input and output alphabet Σ. Let ζ and ξ be dominant elements of $\mathcal{P}^O_{\text{refl}}$ and $\mathcal{M}^O_{\text{refl}}$. respectively. Then there is a stochastic $\pi_S \in \mathcal{P}^O_{\text{refl}}$ for ζ, ξ.

Proof. In this case the conditionals of ξ are O-estimable because it is in $\mathcal{M}^O_{\text{refl}}$, and as before the argument follows Theorem 18. ■

11 Conclusion

We have constructed a policy class meeting the requirements of [KL93a]. This can be seen as a justifying the importance of Nash equilibria from a Bayesian perspective. When a Bayesian player's strategy is private information, we see no convincing reason for him to play a strategy corresponding to any Nash equilibrium.For instance, a Bayesian who has observed that his opponent in a game of "rock, paper, scissors" usually chooses "rock" (in past otherwise similar games) might naturally choose "paper," instead of randomizing uniformly as anticipated by classical game theory. However, when the game and policy classes satisfy the grain of truth property, players eventually converge to a set of strategies close to a ε-Nash equilibrium.

Reflective oracles in the real world. It is interesting to consider whether the grain of truth property is a reasonable assumption about the beliefs and strategies of humans. The limit-computability of (some) reflective oracles makes this at least distantly plausible. Certainly humans should model others as demonstrating roughly comparable computational power to ourselves, and our computational boundedness means any mutual recursion must eventually terminate. Reflective oracles are one possible model of this termination, but it is hard to see how the (critical) assumption that all players use the same reflective oracle can be justified. At least in the self-predictive case it is sensible for an agent to use the same reflective oracle to model their beliefs about themselves and their environment; convergence results should still hold when the true environment does not actually use oracle access, as long as it is in $\mathcal{M}^O_{\text{refl}}$. More speculatively, perhaps a shared reflective oracle is a reasonable assumption for (semi-)cooperative multi-agent systems such as members of a common culture or subagents within a cognitive architecture.

Future work. It would be interesting to determine the degree of centrality and uniqueness of reflective oracles (and the corresponding $\mathcal{P}^O_{\text{refl}}$) among the solutions to the grain of truth problem, and to study in general the connection between the grain of truth problem and self-prediction. Another fascinating research direction is to clarify the computability properties of reflective oracles; we know that there exist limit-computable reflective oracles, but not all reflective oracles are limit-computable because any measure is computable with respect to some reflective oracle (see Appendix B). Therefore, it is easy to understand $\bigcup_O \mathcal{P}^O_{\text{refl}}$. We showed in Lemma 10 that

regardless of the chosen reflective oracle O, all estimable measures ν[15] are sampled by some probabilistic Turing machine with access to O (by binary search), but are any other measures in the intersection $\bigcap_O \mathcal{P}^O_{\text{refl}}$?

12 Acknowledgements

This work was supported in part by a grant from the Long-Term Future Fund (EA Funds - Cole Wyeth Household - 9/26/2023).

References

[Tho33] William R Thompson. "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples". In: *Biometrika* 25.3-4 (1933), pp. 285–294.

[NMR44] John von Neumann, Oskar Morgenstern, and Ariel Rubinstein. *Theory of Games and Economic Behavior (60th Anniversary Commemorative Edition)*. Princeton University Press, 1944. ISBN: 9780691130613. URL: https://www.jstor.org/stable/j.ctt1r2gkx (visited on 03/18/2024).

[Fan52] Ky Fan. "Fixed-point and minimax theorems in locally convex topological linear spaces". In: *Proceedings of the National Academy of Sciences* 38.2 (1952), pp. 121–126.

[Kle52] Stephen Cole Kleene. *Introduction to Metamathematics*. Groningen: P. Noordhoff N.V., 1952.

[Aum64] Robert J . Aumann. "28. Mixed and Behavior Strategies in Infinite Extensive Games". In: *Advances in Game Theory. (AM-52), Volume 52*. Ed. by Melvin Dresher, Lloyd S. Shapley, and Albert William Tucker. Princeton: Princeton University Press, 1964, pp. 627–650. ISBN: 9781400882014. DOI: doi:10.1515/9781400882014-029. URL: https://doi.org/10.1515/9781400882014-029.

[Rud91] W. Rudin. *Functional Analysis*. International series in pure and applied mathematics. McGraw-Hill, 1991. ISBN: 9780070542365. URL: https://books.google.ca/books?id=Sh_vAAAAMAAJ.

[KL93a] Ehud Kalai and Ehud Lehrer. "Rational Learning Leads to Nash Equilibrium". In: *Econometrica* 61.5 (1993), pp. 1019–1045. ISSN: 0012-9682. DOI: 10.2307/2951492. URL: https://www.jstor.org/stable/2951492 (visited on 03/18/2024).

[15] Here we mean that $\nu(x)$ should be estimable, not necessarily the conditional $\nu(\alpha|x)$.

[KL93b] Ehud Kalai and Ehud Lehrer. "Subjective Equilibrium in Repeated Games". In: *Econometrica* 61.5 (1993), pp. 1231–1240. ISSN: 0012-9682. DOI: 10.2307/2951500. URL: https://www.jstor.org/stable/2951500 (visited on 05/24/2024).

[Nac97] John H. Nachbar. "Prediction, Optimization, and Learning in Repeated Games". In: *Econometrica* 65.2 (1997), pp. 275–309. ISSN: 0012-9682. DOI: 10.2307/2171894. URL: https://www.jstor.org/stable/2171894 (visited on 03/18/2024).

[FY01] Dean P. Foster and H. Peyton Young. "On the impossibility of predicting the behavior of rational agents". In: *Proceedings of the National Academy of Sciences* 98.22 (Oct. 2001), pp. 12848–12853. DOI: 10.1073/pnas.211534898. URL: https://www.pnas.org/doi/full/10.1073/pnas.211534898 (visited on 03/18/2024).

[Hut05] Marcus Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. 300 pages, http://www.hutter1.net/ai/uaibook.htm. Berlin: Springer, 2005. ISBN: 3-540-22139-5. DOI: 10.1007/b138233. URL: http://www.hutter1.net/ai/uaibook.htm.

[Nac05] John H. Nachbar. "Beliefs in Repeated Games". In: *Econometrica* 73.2 (2005), pp. 459–480. ISSN: 0012-9682. URL: https://www.jstor.org/stable/3598794 (visited on 03/18/2024).

[LV+08] Ming Li, Paul Vitányi, et al. *An introduction to Kolmogorov complexity and its applications*. Vol. 3. Springer, 2008.

[Ors10] L. Orseau. "Optimality Issues of Universal Greedy Agents with Static Priors". In: *Proc. 21st International Conf. on Algorithmic Learning Theory (ALT'10)*. Vol. 6331. LNAI. Canberra, Australia: Springer, 2010, pp. 345–359. ISBN: 978-3-642-16107-0. DOI: 10.1007/978-3-642-16108-7_28.

[LH14] Tor Lattimore and Marcus Hutter. "General time consistent discounting". In: *Theoretical Computer Science*. Algorithmic Learning Theory 519 (Jan. 2014), pp. 140–154. ISSN: 0304-3975. DOI: 10.1016/j.tcs.2013.09.022. URL: https://www.sciencedirect.com/science/article/pii/S0304397513007135 (visited on 05/22/2024).

[FST15] Benja Fallenstein, Nate Soares, and Jessica Taylor. "Reflective Variants of Solomonoff Induction and AIXI". en. In: *Artificial General Intelligence*. Ed. by Jordi Bieger, Ben Goertzel, and Alexey Potapov. Cham: Springer International Publishing, 2015, pp. 60–69. ISBN: 9783319213651. DOI: 10.1007/978-3-319-21365-1_7.

[FTC15] Benja Fallenstein, Jessica Taylor, and Paul F. Christiano. *Reflective Oracles: A Foundation for Classical Game Theory*. arXiv:1508.04145 [cs]. Aug. 2015. DOI: 10.48550/arXiv.1508.04145. URL: http://arxiv.org/abs/1508.04145 (visited on 03/04/2024).

[LH15] Jan Leike and Marcus Hutter. "Bad Universal Priors and Notions of Optimality". en. In: *Proceedings of The 28th Conference on Learning Theory*. PMLR, June 2015, pp. 1244–1259. URL: https://proceedings.mlr.press/v40/Leike15.html (visited on 07/12/2024).

[LTF16] Jan Leike, Jessica Taylor, and Benya Fallenstein. "A formal solution to the grain of truth problem". In: *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*. UAI'16. Arlington, Virginia, USA: AUAI Press, June 25, 2016, pp. 427–436. ISBN: 9780996643115. (Visited on 07/12/2024).

[Lei+16] Jan Leike et al. "Thompson Sampling is Asymptotically Optimal in General Environments". In: *Proc. 32nd International Conf. on Uncertainty in Artificial Intelligence (UAI'16)*. New Jersey, USA: AUAI Press, 2016, pp. 417–426. ISBN: 978-0-9966431-1-5. URL: http://arxiv.org/abs/1602.07905.

[Cat+23] Elliot Catt et al. "Self-Predictive Universal AI". en. In: *Advances in Neural Information Processing Systems* 36 (Dec. 2023), pp. 27181–27198. URL: https://proceedings.neurips.cc/paper_files/paper/2023/hash/56a225639da77e8f7c0409f6d5ba996b-Abstract-Conference.html (visited on 04/23/2024).

[HQC24] Marcus Hutter, David Quarel, and Elliot Catt. *An Introduction to Universal Artificial Intelligence*. Chapman & Hall/CRC Artificial Intelligence and Robotics Series. Taylor and Francis, 2024, p. 500. ISBN: 9781032607023. URL: http://www.hutter1.net/ai/uaibook2.htm.

[Ale] Samuel Allen Alexander. "Private Memory Confers No Advantage". In: *Cifma* ().

A List of Notation

Our notation follow [LTF16] as closely as possible.

$[\![R]\!]$	$= 1$ if R=true and $=0$ if R=false (Iverson bracket)
$:=$	defined to be equal
$\mathbb{N}$	the natural numbers
$\mathbb{Q}$	the rational numbers
$\mathbb{R}$	the real numbers

t	(current) interaction (time) step, $t \in \mathbb{N}$
i	player=agent index $\in \{1, ..., n\}$
j	the other players=agents $\neq i$ part of the environment
k	natural number indices
n	number of agents=players
p, q	a real value with interpretation as a probability.
$\mathcal{X}^*$	the set of all finite strings over the alphabet $\mathcal{X}$
$\mathcal{X}^\infty$	the set of all infinite sequences over the alphabet $\mathcal{X}$
$\Delta\mathcal{X}$	a probability distribution over $\mathcal{X}$
O	a reflective oracle
$\tilde{O}$	a partial oracle
flip	the function that on input $p \in [0, 1]$ returns 1 with probability p, else 0.
$\mathcal{Q}_i$	a query to an oracle in an enumeration $i \in \mathcal{N}$
$\mathcal{T}$	the set of oracle probabilistic Turing machines, extended to include tape contents
T	an oracle probabilistic Turing machine
λ_T	the semimeasure corresponding to the probabilistic Turing machine T
λ_T^O	the semimeasure corresponding to the probabilistic Turing machine T with access to the reflective oracle O
$\bar{\lambda}_T^O$	the oracle-completed semimeasure corresponding to the probabilistic Turing machine T with access to the reflective oracle O
rO	abbreviation for "reflective oracle"
O-sampled	a semimeasures ν is O-sampled if there exists a pTM T such that $\nu = \lambda_T^O$
O-estimable	æstimable with access to the reflective oracle O
ν	a semimeasure sometimes representing an environment
μ	the true environment
$\mu \ll \nu$	μ is absolutely continuous w.r.t. ν
$\mathcal{A}$	a finite alphabet often identified with the set of possible actions
$\mathcal{E}$	a finite alphabet containing the possible percepts; rewards should be computable from percepts
α, β	alphabet symbols usually in $\mathcal{A}$
a_t	the action(s) in time step t
e_t	the percept(s) in time step t
æ$_{<t}$	the first $t-1$ interactions, $a_1 e_1 ... a_{t-1} e_{t-1}$
ϵ	the empty string
ε	a small positive real number
γ	the discount function $\gamma : \mathbb{N} \to R_{\geq 0}$

Γ_t	a discount normalization factor $\Gamma_t = \sum_{k=t}^{\infty} \gamma_k$
ν, μ	environments/semimeasures
σ	a multi-player game
$\sigma^{\pi_{1:n}}$	history distribution induced by $\pi_1, ..., \pi_n$ interacting in the multi-player game σ
σ_i	the subjective environment of player i in multi-player game σ
π	a policy (strategy), $\pi : (\mathcal{A} \times \mathcal{E})^* \to \mathcal{A}$
V_ν^π	the ν-expected value of policy π
V_ν^*	the optimal value in environment ν
$\mathcal{G}$	a countable class of multi-player games
$\mathcal{M}$	a countable class of environments
$\mathcal{M}_{\text{refl}}^O$	the class of environments computed by probabilistic Turing machines with access to reflective oracle O
$\mathcal{P}$	a countable class of policies (strategies)
$\mathcal{P}_{\text{refl}}^O$	the class of policies (strategies) computed by probabilistic Turing machines with access to reflective oracle O
w	a real valued weight distribution, usually over $\mathcal{M}_{\text{refl}}^O$ or $\mathcal{P}_{\text{refl}}^O$
ξ	a mixture environment, usually over $\mathcal{M}_{\text{refl}}^O$
ζ	a mixture policy over $\mathcal{P}_{\text{refl}}^O$

B Non-binary Alphabet Reflective Oracles

We provide a detailed existence proof for non-binary reflective oracles. First, we extend the definition of reflective oracles to the non-binary case, which requires slightly stronger conditions than the binary case to ensure that the oracle completed conditionals lie on a probability simplex. Then we construct a point-to-set mapping that expresses an oracle's self-consistency and show that a fixed point would imply the existence of a reflective oracle. Finally, we demonstrate the existence of a fixed point using the infinite dimensional version of the Kakutani fixed point theorem [Fan52].

Definition. The most important feature of a reflective oracle is that it can be used to complete a semi-measure to a measure (which is then reflective-oracle sampled and even estimable). Extending the definition to non-binary alphabets, we must take care to preserve this property. It is easiest to do this by extending the strict definition of Fallenstein et. al. [FTC15], which we refer to as a "step reflective oracle." Slightly generalizing their notation to explicitly include the input $x \in \{0,1\}^*$, they require that for every machine T and input x, there is some cutoff $P[T(x) = 1] \leq q \leq 1 - P[T(x) \neq 1]$ such that

$$p < q \Rightarrow O(T, x, p) = 1$$

$$p > q \Rightarrow O(T, x, p) = 0$$

We will generalize to finite output alphabet $\mathcal{A}$ and input alphabet Σ. For our purposes $\mathcal{A} \subset \Sigma$; for instance $\mathcal{A}$ is a set of actions and Σ includes actions and observations.

Definition 32 (reflective oracle) The oracle O valued in [0,1] is reflective iff for each probabilistic TM T and string $x \in \Sigma^*$, $\exists \{q_\alpha\}_{\alpha \in \mathcal{A}}$ satisfying the following properties:

$$\sum_{\alpha \in \mathcal{A}} q_\alpha \;=\; 1$$

And for all $\alpha \in \mathcal{A}$,

$$P[T^O(x) = \alpha] \;\leq\; q_\alpha \;\leq\; 1 - P[T^O(x) \neq \alpha]$$

$$p < q_\alpha \;\Rightarrow\; O_\alpha(T, x, p) = 1$$

$$p > q_\alpha \;\Rightarrow\; O_\alpha(T, x, p) = 0$$

Notice that oracles are now indexed by α, so can be thought of as a family satisfying certain relations or as accepting a new argument of type $\mathcal{A}$. An oracle responds to queries with 0 or 1, and the value of O_α on a given query is interpreted as the probability that its answer is 1. Given this definition, it is not difficult to see how to conduct a binary search for each q_α and recover a measure.

Proof. Now we must prove that non-binary reflective oracles actually exist. The argument follows the original existence proof. The key is to restrict certain functions to lie on the simplex (which is closed, compact, and convex) which makes it possible to ensure that the point-set map only accepts and produces functions satisfying the first condition.

The argument relies on the infinite dimensional version of the Kakutani fixed point theorem (sometimes more appropriately but less helpfully referred to as the Kakutani-Ky Fan theorem), which I'll reproduce from Ky Fan's paper [Fan52]:

Theorem 33 (infinite dimensional Kakutani fixed point theorem) Let L be a locally convex topological linear space and K a compact convex set in L. Let $R(K)$ be the family of all closed convex (non-empty) subsets of K. Then for any upper semicontinous point-to-set transformation f from K into $R(K)$, there exists a point $x_0 \in K$ s.t. $x_0 \in f(x_0)$.

The more modern term for locally convex topogological linear space is locally convex topological vector space (LCTVS).

Let $\mathcal{T}$ be the space of pTM's generalized to include the tape configuration. We will consider points

$$(\text{query}, \text{eval})$$

where eval $\in (\Delta\mathcal{A})^{\mathcal{T}}$, which we interpret as (for a fixed point) returning the completed chance of a given pTM outputting α, and query $\in [0,1]^{\mathcal{T} \times (\mathbb{Q} \cap [0,1]) \times \mathcal{A}}$ which is a function

on queries returning the oracle's chance of producing a 1. We will later introduce constraints on query in terms of eval (through the point-to-set map f) so that a fixed point is a reflective oracle, which is why the definition of the space for query does not need to enforce that the oracle's answers describe a probability distribution. The space of (query, eval) pairs is

$$K = [0,1]^{\mathcal{T}\times(\mathbb{Q}\cap[0,1])\times\mathcal{A}} \times (\Delta\mathcal{A})^{\mathcal{T}}$$

This is a subset (under a trivial "currying" homeomorphism on the space of eval) of

$$S = \mathbb{R}^{\mathcal{T}\times(\mathbb{Q}\cap[0,1])\times\mathcal{A}} \times \mathbb{R}^{(\mathcal{T}\times\mathcal{A})}$$

which is an LCTVS under the product topology. This follow from the fact that $\mathbb{R}$ is a (very simple) LCTVS, and all LCTVS properties are closed under the product operation.

Topological properties. We will later need that S is metrizable to use a notion of sequential convergence instead of "upper semicontinuity". Because $\mathbb{R}$ is Hausdorff and products of Hausdorff spaces are also Hausdorff, S is a LCTVS in the stricter sense of Rudin's Functional Analysis [Rud91] (which requires T_1, a property weaker than Hausdorff). Theorem 9 tells us that TVS is metrizable if it has a countable local base. Our countable local base for S is a sequence of balls with radius $1/n$ at the first n points in enumerations of both the exponents ($\mathcal{T}\times\mathcal{A}$ and $\mathcal{T}\times(\mathbb{Q}\cap[0,1])\times\mathcal{A}$), with $\mathbb{R}$ at the (infinitely many) remaining points. So S is metrizable. Now we need to know that K is compact and convex. Tychonoff's theorem implies it is compact. Convexity is immediate from convexity of simplices.

The point-to-set map f. The original proof demonstrates that it is possible to construct an oracle reflective on some subset of queries and equal to a different fixed oracle elsewhere. This is more than we need and clutters the proof, but is easy to understand once the proof is digested, so we focus on the case that we want a reflective oracle on all queries. We are ready to define our point-to-set map $f : K \to 2^K$. We will then demonstrate that the range of f is $R(K)$. Let $(\text{query}, \text{eval}) \in f((\text{query}, \text{eval}))$ iff the following conditions hold:

$$p < \text{eval}_\alpha(T) \Rightarrow \text{query}'_\alpha(T,p) = 1$$

$$p > \text{eval}_\alpha(T) \Rightarrow \text{query}'_\alpha(T,p) = 0$$

This leaves $\text{query}'_\alpha(T, \text{eval}_\alpha(T))$ unconstrained.

The recursive rules on eval$'$ are slightly more complicated. The "base case" is that if T returns $T() = \beta$ on its next computation step,

$$\text{eval}'_\alpha(T) = \delta_\alpha(T()) = [\![\alpha = \beta]\!]$$

T can halt returning nothing or something other than a single symbol. In that case, $\text{eval}'_\alpha(T)$ is arbitrary.

The other "inductive" rules are copied from [FTC15]. If T performs a deterministic computation step producing a new machine/configuration N, $\text{eval}'_\alpha(T) = \text{eval}(N)$. If T performs a coin flip yielding a state N with rational probability p and N' with rational probability $1-p$, $\text{eval}'_\alpha(T) = p\,\text{eval}_\alpha(N) + (1-p)\,\text{eval}_\alpha(N')$. We observe that typically a transition of a pTM is defined to use exactly one random bit so we should have $p = \frac{1}{2}$. Also, if T has a chance of halting with some output immediately after reading the random bit (in the same computation step), $\text{eval}_\alpha(N)$ or $\text{eval}_\alpha(N')$ should be replaced according to the first rule; otherwise they are not really well-defined. Calls to the oracle should be treated the same as coin flips with probability determined by query: if the oracle is invoked on (T,p) yielding N on 1 and N' on 0, and $\text{query}(T,p) = q$ then $\text{eval}'_\alpha(T) = q\,\text{eval}_\alpha(N) + (1-q)\,\text{eval}_\alpha(N')$.

Assuming for a moment that a fixed point exists, we will show that this gives us a reflective oracle, defined by $P[O_\alpha(T,p) = 1] = \text{query}_\alpha(T,p)$. We normally want our reflective oracles to accept a machine description and input string, but these can be combined into an extended machine/configuration as in the definition of O.

By induction on the number of computation steps,

$$P[T^O() = \alpha] \leq \text{eval}_\alpha(T) \leq 1 - P[T^O() \neq \alpha]$$

Together with the conditions of $\text{query}_\alpha(T,p)$, this shows that $\text{eval}_\alpha(T)$ satisfies the last three conditions on q_α in Appendix B. The first condition is automatically satisfied because $\text{eval}(T)$ is restricted to the simplex; this pushes the burden of proof to the non-emptiness of $f((\text{query}, \text{eval}))$. Therefore, O is a reflective oracle.

Existence of a fixed point. Now we only need to prove the existence of a fixed point by establishing the conditions of the infinite dimensional version of the Kakutani fixed-point theorem. First, we will show that $f((\text{query}, \text{eval}))$ is closed, convex, and non-empty. In a metrizable space, closed sets can be characterized in the ordinary way by sequential convergence. Note that at every point, query' and eval' are either restricted to some fixed value depending on $(\text{query}, \text{eval})$ or are unrestricted (except for eval' to lie on the simplex, which is closed). Taking limits (of any such sequence) shows that $f((\text{query}, \text{eval}))$ is closed. Convexity is also easy to verify pointwise (and from convexity of the simplex). Finally, we must show non-emptiness. It is obvious that the conditions on query' can always be satisfied. It remains only to show that the conditions on eval' can be satisfied by points lying on the simplex. But this is only another application of convexity of the simplex, noting that q and p lie in [0,1], and $\text{eval}(T) \in \Delta\mathcal{A}$, and verifying that the "base case" condition produces a point on the simplex (in fact an extreme point of the simplex).

The last property to verify is "upper semicontinuity." But as Fan points out, this is equivalent to the definition in terms of convergent sequences, which is usually called the "closed graph property." We must show that if $(\text{query}'_n, \text{eval}'_n) \to (\text{query}', \text{eval}')$, $(\text{query}_n, \text{eval}_n) \to (\text{query}, \text{eval})$, and $(\text{query}'_n, \text{eval}'_n) \in f((\text{query}_n, \text{eval}_n))$, then $(\text{query}', \text{eval}') \in f((\text{query}, \text{eval}))$.

The argument is exactly the same as in the binary case, but applies "pointwise" to each α. Taking limits on both sides of the rules for eval$'$ immediately gives the desired result for this part. For $\text{query}'_\alpha(T,p)$, the argument depends on the relationship between $\text{eval}_\alpha(T)$ and p. If they are equal, the condition is automatically satisfied. The two remaining cases are symmetric, so we consider (w.l.o.g.) the case that $\text{eval}_\alpha(T) > p$. Since $\text{eval}_n \to \text{eval}$ (in the topology of pointwise convergence), $(\text{eval}_n)_\alpha(T) \to \text{eval}_\alpha(T)$, and for sufficiently large n, $(\text{eval}_n)_\alpha(T) > p$. This means that $(\text{query}'_n)_\alpha(T,p) \to 1$, so $\text{query}'_\alpha(T,p) = 1$. This proves the closed graph property, which implies a fixed point of f by the infinite dimensional Kakutani fixed point theorem. Therefore, there is a non-binary alphabet reflective oracle.

Types for symbols. Sometimes the alphabet $\mathcal{A} = \bigsqcup_{1\le i\le n} \mathcal{A}_i$ and each machine has an intended *type* in $\{1,..,n\}$. Then we want to interpret our machines as semimeasures over the corresponding $\mathcal{A}_i^*$ and complete the conditionals to $\Delta\mathcal{A}_i$. The primary examples are when each player has a different action set (so n is the number of players) and when a player interacts with an environment (so $n = 2$ for the action and percept spaces). The natural idea is to use a different reflective oracle with each output alphabet $\mathcal{A}_i$, but unfortunately the oracles need to be answer questions about each other's behavior so this does not literally work. Instead we can change the requirement $\sum_{\alpha\in\mathcal{A}} q_\alpha = 1$ to the n requirements $\sum_{\alpha\in\mathcal{A}_i} q_\alpha = 1$, and force the oracle to satisfy this by changing the space of eval from $(\Delta\mathcal{A})^{\mathcal{T}}$ to $(\prod_{i=1}^n \Delta\mathcal{A}_i)^{\mathcal{T}}$. It is easy to see that this set is still closed, convex, and compact. The rest of the existence proof goes through essentially unchanged. The resulting (typed) non-binary oracle automatically knows which output type we expect for a machine based on the type of its symbol argument $\alpha \in \mathcal{A}_i$ and completes the associated semimeasure appropriately, redistributing the probability of outputs outside of $\mathcal{A}_i$ in the same way as non-halting probability mass.

Reflectivity on subsets. It is easy to modify the proof above so that we construct an oracle O that satisfies reflectivity on some subset of queries R but behaves identically to any arbitrary oracle O' outside of R. Let A be a *closed* set of pTM's that only make oracle calls about other pTM's in A. Let $R = \{(T,x,p) | T \notin A\}$. Then if O' is reflective on queries about pTM's in A (that is, R^C), we can construct an oracle O reflective on R that agrees with O' on R^C. But because replacing O' by O does not change the results of any oracle calls for machines in A, O is also reflective on R^C, which means O is a reflective oracle (on all queries). This is analogous to the extending a linearly indepedent set of vectors to a basis. In some cases it is easy to find a variety of explicit oracles O' reflective on A; for instance, if $A = \{T\}$ where T never halts, O' can complete λ_T^O to a measure in any arbitrary way, and there will exist a reflective oracle O agreeing with O' about T. This means that all measures are reflective-oracle computable with appropriate choice of O (though by countability of $\mathcal{T}$ there is no *particular* O that makes every measure rO-computable).

C Limit-Computable Step Reflective Oracles

We will extend Leike's proof that there is a limit-computable reflective oracle [LTF16] to show that there is a limit-computable non-binary reflective oracle. This requires a slightly different construction which yields a limit-computable step reflective oracle when restricted to the binary case, simplifying the oracle completion process by removing the need for an expectation.

Following [LTF16], we will construct an infinite sequence of partial oracles converging to a reflective oracle in the limit. The set of queries to a reflective oracle is countable and computably enumerable, so we will fix a computable enumeration:

$$\mathcal{T} \times \Sigma^* \times \mathbb{Q} =: \{\mathcal{Q}_1, \mathcal{Q}_2, ...\}$$

where $\mathcal{T}$ is the set of (generalized) pTM's as above and Σ is the input alphabet. A reflective oracle is also indexable by symbols from the output alphabet.

Definition 34 (k-partial oracle) A k-partial oracle $\tilde{O}_\alpha$ is a function from the first k queries to the multiples of 2^{-k} in [0,1]:

$$\tilde{O}_\alpha : \{\mathcal{Q}_1, \mathcal{Q}_2, ..., \mathcal{Q}_k\} \to \{n2^{-k} | 0 \leq n \leq 2^k\}$$

Definition 35 (approximate an oracle) A k-partial oracle $\tilde{O}$ approximates an oracle O iff $\forall \alpha | O_\alpha(\mathcal{Q}_i) - \tilde{O}_\alpha(\mathcal{Q}_i)| \leq 2^{-k-1}$ for all $i \leq k$.

Let $\tilde{O}$ be a k-partial oracle for $k \in \mathbb{N}$ and let $T \in \mathcal{T}$ be an oracle machine. We define $T^{\tilde{O}}$ to be the following machine:

1. Run T for at most k steps.
2. If T makes an oracle α call on $\mathcal{Q}_i$ for $i \leq k$,
   (a) Return 1 with probability $\tilde{O}_\alpha(\mathcal{Q}_i) - 2^{-k}$
   (b) Return 0 with probability $1 - \tilde{O}_\alpha(\mathcal{Q}_i) - 2^{-k}$
   (c) halt otherwise
3. If T calls the oracle on $\mathcal{Q}_j$ for $j > k$, halt.

Since $\tilde{O}$ is not a fully defined oracle, this is different than the usual meaning of T^O. In particular it implicitly depends on the value of k.

Lemma 36 (bound on $\boldsymbol{\lambda_T^{\tilde{O}}}$) If a k-partial oracle $\tilde{O}$ approximates a reflective oracle O, then $\lambda_T^O(\alpha|x) \geq \lambda_T^{\tilde{O}}(\alpha|x)$ for all $\alpha \in \mathcal{A}$, $x \in \Sigma^*$, and $T \in \mathcal{T}$.

Proof. This follows from the definition of $T^{\tilde{O}}$: when running T with $\tilde{O}$ instead of O, every sequence of oracle responses is less likely because $\tilde{O}_\alpha - 2^{-k} < O_\alpha(\mathcal{Q}_i)$ and $1 - \tilde{O}_\alpha - 2^{-k} < 1 - O(\mathcal{Q}_i)$. The other differences can also only lose probability mass. If T makes calls whose index is $> k$ or runs for more than k steps the machine halts and no output is generated. ■

Definition 37 (*k*-partially reflective) A k-partial oracle $\tilde{O}$ is k-partially (step) reflective iff for the first k queries (T, x, p) and for all α

1. $p < \lambda_T^{\tilde{O}}(\alpha|x)$ implies $\tilde{O}_\alpha(T, x, p) = 1$, and
2. $p > 1 - \sum_{\beta \neq \alpha} \lambda_T^{\tilde{O}}(\beta|x)$ implies $\tilde{O}_\alpha(T, x, p) = 0$.

Also, we require that for all α, T, x, $\tilde{O}_\alpha(T, x, \cdot)$ is non-increasing and there is at most one p such that (T, x, p) is in the first k queries and $\tilde{O}_\alpha(T, x, p) \notin \{0, 1\}$.

Finally, for each T, x appearing in one of the first k queries,

1. $\sum_{\alpha \in \mathcal{A}} \min\{p | (T, x, p) \in \{Q_i\}_{i \leq k} \text{ and } \tilde{O}_\alpha(T, x, p) = 0\} \geq 1$
2. $\sum_{\alpha \in \mathcal{A}} \max\{p | (T, x, p) \in \{Q_i\}_{i \leq k} \text{ and } \tilde{O}_\alpha(T, x, p) = 1\} \leq 1$

The minima default to 1 and the maxima default to 0. The first pair of conditions enforce the reflective oracle property. The remaining conditions enforce the step reflective oracle property which is always required for non-binary reflective oracles.

We can check whether a k-partial oracle is k-partially reflective in finite time. The first pair of conditions can be checked by running the machines from the first k queries for k steps each (on every combination of the $\leq 2^k$ random bits used) and calculating $\lambda_T^{\tilde{O}}(\alpha|x)$ exactly. The rest are clearly possible to verify with one pass over the first k queries for each α.

Lemma 38 (partial approximations are partially reflective) If O is a reflective oracle and $\tilde{O}$ is a k-partial oracle that approximates O, then $\tilde{O}$ is k-partially reflective.

Proof. Note that since $\tilde{O}$ assigns values in a 2^{-k} grid and approximates O up to 2^{-k-1}, for the first k queries $O_\alpha(T, x, p) = 0 \to \tilde{O}_\alpha(T, x, p) = 0$ and $O_\alpha(T, x, p) = 1 \to \tilde{O}_\alpha(T, x, p) = 1$. Assuming $\lambda_T^{\tilde{O}}(\alpha|x) > p$ we get from Lemma 36 that $\lambda_T^{O}(\alpha|x) \geq \lambda_T^{\tilde{O}}(\alpha|x) \geq p$ so $1 = O_\alpha(T, x, p) = \tilde{O}_\alpha(T, x, p)$. The second condition is proved symmetrically. For the remaining conditions, recall that for each T, x, $\exists q_\alpha$ with $O(T, x, p) = 1$ for $p < q_\alpha$, $O(T, x, p) = 0$ for $p > q_\alpha$, and $\sum_\alpha q_\alpha = 1$. This means that $\tilde{O}_\alpha(T, x, p) = 1$ for $p < q_\alpha$ and $\tilde{O}_\alpha(T, x, p) = 0$ for $p > q_\alpha$, and since these are the maximum and minimum values of $\tilde{O}$ it is certainly non-increasing regardless of its value at q_α (which is in general not defined). Additionally, if $\tilde{O}_\alpha(T, x, p) = 0$ then certainly $p \geq q_\alpha$, which implies that the sum of minima is $\geq \sum_\alpha q_\alpha = 1$ (this is also automatically satisfied if any $\tilde{O}_\alpha(T, x, \cdot)$ does not take the value 0). The argument for the bound on the maxima is similar. ■

Definition 39 (extending partial oracles) A $k+1$ partial oracle $\tilde{O}'$ extends a k-partial oracle $\tilde{O}$ iff $|\tilde{O}_\alpha(\mathcal{Q}_i) - \tilde{O}'_\alpha(\mathcal{Q}_i)| \leq 2^{-k-1}$ for all $i \leq k$.

Lemma 40 (infinite sequence of extensions) There is an infinite sequence of partial oracles $(\tilde{O}^k)_{k \in \mathbb{N}}$ such that for each k, $\tilde{O}^k$ is a k-partially reflective k-partial oracle and $\tilde{O}^{k+1}$ extends $\tilde{O}^k$.

Proof. As shown in Appendix B, there is a (step) reflective oracle O for any finite alphabet. For every k, there is a canonical k-partial oracle $\tilde{O}_k$ that approximates O: restrict O to the first k queries and for any such query $\mathcal{Q}$ for each $\alpha \in \mathcal{A}$ pick the value in the 2^{-k} grid which is closest to $O_\alpha(\mathcal{Q})$. By construction, each $\tilde{O}^{k+1}$ extends $\tilde{O}^k$ and by Lemma 38, each $\tilde{O}^k$ is k-partially reflective. ■

Lemma 41 ($\boldsymbol{\lambda_T^{\tilde{O}^k}}$ increases) If the $k+1$ partial oracle $\tilde{O}^{k+1}$ extends the k-partial oracle $\tilde{O}_k$, then $\forall \alpha\ \lambda_T^{\tilde{O}^{k+1}}(\alpha|x) \geq \lambda_T^{\tilde{O}^k}(\alpha|x)$ for each $T \in \mathcal{T}$ and $x \in \mathcal{A}^*$

Proof. $T^{\tilde{O}^{k+1}}$ runs for one more step than $T^{\tilde{O}^k}$ and can answer one more query. Because $\tilde{O}^{k+1}$ extends $\tilde{O}^k$, $|\tilde{O}^{k+1}_\alpha(\mathcal{Q}_i) - \tilde{O}^k_\alpha(\mathcal{Q}_i)| \leq 2^{-k-1}$, which means $\tilde{O}^{k+1}_\alpha(\mathcal{Q}_i) - 2^{-k-1} \geq \tilde{O}^k_\alpha(\mathcal{Q}_i) - 2^{-k}$, so halting on oracle calls is less likely and the chances of returning 0 or 1 are both higher. ■

Search algorithm. Now we are prepared to state the algorithm that constructs a reflective oracle in the limit. The algorithm recursively traverses a directed acyclic graph (DAG) of partial oracles. The DAG's nodes are the partial oracles; level k of the DAG contains all k-partial oracles. There is an edge in the DAG from the k-partial oracle $\tilde{O}^k$ to the i-partial oracle $\tilde{O}^i$ if and only if $i = k+1$ and $\tilde{O}^i$ extends $\tilde{O}^k$.

For every k, there are only finitely many k-partial oracles, since they are functions from finite sets to finite sets. In particular, there are exactly two 1-partial oracles (so the DAG has two *sources*, nodes without parents where the search can begin). Pick one of them to start with, and proceed recursively as follows. Given a k-partial oracle $\tilde{O}^k$, there are finitely many $(k+1)$-partial oracles that extend $\tilde{O}^k$ (finite out-degree). Pick one that is $(k+1)$-partially reflective (which can be checked in finite time). If there is no $(k+1)$-partially reflective extension, backtrack.

By Lemma 40 our DAG is infinitely deep and thus the search does not terminate. Moreover, it can backtrack to each level only a finite number of times because there are only a finite number of paths from a source to each level and at each level there are only a finite number of possible extensions (in fact, though it is possible for two different paths from sources in the DAG to reach the same node, there is no need to return to any node once it has been visited and all of its children have been explored). Therefore, the algorithm will produce an infinite sequence of partial oracles, each extending the previous. Because of finite backtracking, the output eventually stabilizes on a sequence of partial oracles $\tilde{O}^1, \tilde{O}^2, ...$ By the following lemma, this sequence converges to a reflective oracle, proving Theorem 43.

Lemma 42 (limit is reflective) Let $\tilde{O}^1, \tilde{O}^2, ...$ be a sequence where $\tilde{O}^k$ is a k-partially reflective oracle and $\tilde{O}^{k+1}$ extends $\tilde{O}^k$ for all $k \in \mathbb{N}$. Let $O := \lim_{k\to\infty} \tilde{O}^k$ be the pointwise limit. Then

1. $\lambda_T^{\tilde{O}^k}(\alpha|x) \to \lambda_T^O(\alpha|x)$ as $k \to \infty$ for all $\alpha \in \mathcal{A}$ and $x \in \Sigma^*$.
2. O is a reflective oracle.

Proof. First note that the pointwise limit must exist because $|\tilde{O}^k_\alpha(\mathcal{Q}_i) - \tilde{O}^{k+1}_\alpha(\mathcal{Q}_i)| \leq 2^{-k-1}$ by Definition 39.

1. Since $\tilde{O}^{k+1}$ extends $\tilde{O}^k$, each $\tilde{O}^k$ approximates O. Let $\alpha \in \mathcal{A}, T \in \mathcal{T}$, and $x \in \Sigma^*$ and consider the sequence $a_k := \lambda_T^{\tilde{O}^k}(\alpha|x)$. By Lemma 41, $a_k \leq a_{k+1}$ so the sequence is monotonically increasing. It is also bounded above by $\lambda_T^O(\alpha|x)$ according to Lemma 36, so it converges. It only remains to show that the sequence does not converge to something less than $\lambda_T^O(\alpha|x)$. The probability that T^O halts, which is bounded above by 1, is the sum of its probabilities of halting at each step. Therefore, the probability T^O halts after running for more than k steps is a tail sum that approaches 0 as $k \to \infty$. The distributions on the results of calls to the partial oracles converge to the distribution on the results of calls to O by definition of O and because $2^{-k} \to 0$ in the definition of $T^{\tilde{O}^k}$. The definition of T^O therefore implies that $a_k \to \lambda_T^O(\alpha|x)$ as desired.

2. By definition, O is an oracle. It only remains to show that O satisfies the step reflective conditions given in Appendix B. Consider a fixed $T \in M$ and $x \in \Sigma^*$. Let $P^k = \{p \in \mathbb{Q} | (T, x, p) \in \{\mathcal{Q}_1, \mathcal{Q}_2, ...\mathcal{Q}_k\}\}$. Let L^k_α be the subset of P^k for which $\tilde{O}^k_\alpha(T, x, p) = 1$ and H^k_α be the subset of P^k for which $\tilde{O}^k_\alpha(T, x, p) = 0$. Because $\tilde{O}^k$ is k-partially reflective, there may exist at most one point p_α which is in P^k but is not in L^k_α or H^k_α. Because $\tilde{O}^k$ takes values on a 2^{-k} grid and $\tilde{O}^{k+1}$ extends $\tilde{O}^k$, p_α is not in L^t_α or H^t_α for any $k' \geq k$; it is not possible to reach 0 or 1 from $\tilde{O}^k_\alpha(T, x, p_\alpha)$ by extension since

$$|\tilde{O}^{k'}_\alpha(T, x, p_\alpha) - \tilde{O}^k_\alpha(T, x, p_\alpha)| \leq \sum_{i=k}^{k'-1} |\tilde{O}^{i+1}_\alpha(T, x, p_\alpha) - \tilde{O}^i_\alpha(T, x, p_\alpha)| \leq \sum_{i=k}^{k'-1} 2^{-i-1} < \sum_{i=k}^{\infty} 2^{-i-1} = 2^{-k}$$

This means that if any such p_α exists it does not depend on k, and we will consider only t sufficiently large so that p_α is in P^k but not in L^k_α or U^k_α for all $k \geq t$. We will assume without loss of generality that p_α exists; otherwise the proof is routinely simplified. By uniqueness, no point other than p_α can lie in $P^k - (L^k_\alpha \cup H^k_\alpha)$ for any such k. Using the bounds on extensions again, $L^k_\alpha \subseteq L^{k+1}_\alpha$ and $H^k_\alpha \subseteq H^{k+1}_\alpha$. Noting that k-partial step reflectivity of each $\tilde{O}^k$ requires it is non-increasing, we must have $L^k_\alpha < H^k_\alpha$ element-wise, so $\sup \bigcup_{k \geq t} L^k_\alpha \leq \inf \bigcup_{k \geq t} H^k_\alpha$. Additionally, $\bigcup_{k \geq t} P^k = \mathbb{Q} \cap [0, 1]$ since all queries are enumerated. This implies that $(\bigcup_{k \geq t} L^k_\alpha) \cup (\bigcup_{k \geq t} H^k_\alpha) = \mathbb{Q} \cap [0, 1] - \{p_\alpha\}$. Therefore,

$$\lim_{k \to \infty} \max L^k_\alpha = \sup \bigcup_{k \geq t} L^k_\alpha = \inf \bigcup_{k \geq t} H^k_\alpha = \lim_{k \to \infty} \min H^k_\alpha$$

The non-increasing property also requires that these limits are p_α (in the case that p_α does not exist we use this as its definition). Because H^k_α are increasing sets and eventually include all $p < p_\alpha$, for such p, $1 = \lim_{k \to \infty} \tilde{O}^k(T, x, p) =$

$O(T, x, p)$. Similarly, $O(T, x, p) = 0$ for $p > p_\alpha$. It is also clear that $\lambda_T^O(\alpha|x) \leq p_\alpha \leq 1 - \sum_{\beta \neq \alpha} \lambda_T^O(\beta|x)$ because the bounds are the limits of $\lambda_T^{\tilde{O}^k}(\alpha|x)$ and $1 - \sum_{\beta \neq \alpha} \lambda_T^{\tilde{O}^k}(\beta|x)$ and $\tilde{O}^k$ is k-partially reflective. All that remains to show is that $\sum_\alpha p_\alpha = 1$. But the last pair of requirements for k-partial reflectivity are that $\sum_\alpha \min H_\alpha^k \geq 1$ and $\sum_\alpha \max L_\alpha^k \leq 1$. Taking the limits of both sides, $1 \leq \sum_\alpha p_\alpha \leq 1$, so $\sum_\alpha p_\alpha = 1$. Therefore, O is a reflective oracle.

■

Theorem 43 (limit-computable non-binary reflective oracle) There is a limit-computable reflective oracle over any finite output alphabet.

Proof. The search algorithm produces a reflective oracle in the limit by Lemma 42. ■

D General Reflective Oracle Computability of Completed Semimeasures

We have restricted our focus to step reflective oracles, but [LTF16] uses a more general class (defined only for binary alphabets) that can randomize at any point between $\lambda_T^O(1|x)$ and $1 - \lambda_T^O(0|x)$. The oracle no longer needs to be indexed by symbol because the completed probability of 0 can be determined from the completed probability of 1. With this definition, there is not necessarily a unique crossover point q where $O(T, x, \cdot)$ switches from 1 to 0. However, $\bar{\lambda}_T^O(\cdot|x)$ can still be defined by running a binary search. The only complication is that the limit is no longer deterministic.

Algorithm 4 pTM C_T

Input: $x \in \{0,1\}^*$
Require: Random bit sequence ω
Output: $y \sim \bar{\lambda}_T^O(\cdot|x)$

```
l, h = 0, 1
for i = 1, 2, ... do
    m = (l+h)/2
    if flip(O(T, x, m)) then l ← m
    else h ← m
    if ω_{1:i} + 2^{-i} < l then Return 1
    else if ω_{1:i} > h then Return 0
```

Let $\bar{\lambda}_T^O = \lambda_{C_T}^O$, defined in 4. On input string $x \in \{0,1\}^*$, let p^* be the limit point of the binary search using $O(T, x, \cdot)$. This is a random variable depending on the stochasticity of queries to O. Fixing the oracle's random choices, p^* is the probability (over random bits ω) that C_T returns 1. This implies that the overall probability that

C_T returns 1 is $\bar{\lambda}_T^O = \lambda_{C_T}^O = \mathbb{E}[p^*]$ (with the expectation over the oracle's responses). Because C_T halts with probability 1, $\bar{\lambda}_T^O$ is a measure and is O-estimable (this time with a deterministic binary search).

E Lower Semicomputability of pTM sampled Semimeasures

Theorem 44 (l.s.c. of pTM semimeasures) For any pTM T, λ_T has l.s.c. conditionals.

Proof. We can see this by constructing a binary tree with each edge corresponding to the next random bit received by T. For some sequences of random bits, T halts after reading a prefix without requesting any further bits. If we mark the leaves of this tree with the output of T when it halts (which is deterministic once the bits are fixed) then $\lambda_T(\alpha|x)$ is the sum of the probabilities for each random string along a path from the root to a leaf labeled with α, which is 2^{-l} for a path of length l. Though there are uncountably many infinite sequences of random bits, so we cannot literally run T on all random bit sequences in parallel, we can run a breadth-first search on the binary tree to increasing depths and sum the probabilities of each leaf marked α encountered, so λ_T is l.s.c. ■

Theorem 45 (l.s.c. conditionals are sampled by a pTM) If μ has l.s.c. conditionals, we can find a pTM T such that $\mu = \lambda_T$.

Proof. Let $\phi_\alpha(x, k)$ lower semicompute $\mu(\alpha|x)$ with O access. The trick (similar to the proof of the coding theorem [LV+08]) is to partition the interval into a list P of subintervals, each labeled with a symbol from $\mathcal{A}$. We will define some subroutines for manipulating P by adding a new subinterval on the right of a given length and checking whether a point is in a labeled subinterval. See Algorithm 7 for details.

Algorithm 5 push

Input: P, $\alpha \in \mathcal{A}$, $\Delta \in \mathbb{Q}$

Effect: A subinterval of label α and length Δ is added to P

```
left ← P[-1].right
P.append(label: α, right: left + Δ)
```

Algorithm 6 check

Input: P and ω

Effect: If ω is in a subinterval, return its label

```
left ← 0
for each label, right ∈ P do
    if left ≤ ω ≤ right then
        return label
    left ← right
```

Algorithm 7 sample

Input: ϕ_α, x

Require: random stream ω

Output: $\alpha \sim \mu(\cdot|x)$

```
Let P be an empty list
ψ_α ← 0
for k ← 1 to ∞ do
    for α ∈ 𝒜 do
        Δ ← φ_α(x, k) − ψ_α
        ψ_α ← φ_α(x, k)
        push(P,α,Δ)
    check(P,ω_{1:k})
```

Note that $w_{1:k}$ is only specified to precision 2^{-k}. We extend the $\leq$ and $\geq$ comparisons against it to only succeed when they succeed in the worst case. It is fairly easy to see that in the limit the total area of the partitions for each α is equal to $\mu(\alpha|x)$. Because μ is only assumed to be a semimeasure, it is possible that some of the interval is unallocated and in this case Algorithm 7 may not halt. ■

F The Subjective Environment is Well-Defined

For the subjective environment σ_i to qualify as a true environment, it must not depend on the strategy π_i. With a bit of algebra we can show that it depends on σ and π_j for $j \neq i$ but not π_i:

$$\sigma_i(e_T^i|æ_{<T}^i a_T^i) = \frac{\sum_{æ_{\leq T}^j,\ j\neq i} \sigma^\pi(æ_{\leq T})}{\sum_{æ_{<T}^j a_T^j,\ j\neq i} \sigma^\pi(æ_{<T}a_T)}$$

$$= \frac{\sum_{æ_{\leq T}^j,\ j\neq i} \prod_{t=1}^T \sigma^\pi(e_t|æ_{<t}a_t) \prod_{t=1}^T \sigma^\pi(a_t|æ_{<t})}{\sum_{æ_{<T}^j a_T^j,\ j\neq i} \prod_{t=1}^{T-1} \sigma^\pi(e_t|æ_{<t}a_t) \prod_{t=1}^T \sigma^\pi(a_t|æ_{<t})}$$

$$= \frac{\sum_{æ_{\leq T}^j,\ j\neq i} \prod_{t=1}^T \sigma(e_t|æ_{<t}a_t) \prod_{t=1}^T \prod_{j=1}^n \pi_j(a_t^j|æ_{<t}^j)}{\sum_{æ_{<T}^j a_T^j,\ j\neq i} \prod_{t=1}^{T-1} \sigma(e_t|æ_{<t}a_t) \prod_{t=1}^T \prod_{j=1}^n \pi_j(a_t^j|æ_{<t}^j)}$$

$$= \frac{\prod_{t=1}^T \pi_i(a_t^i|æ_{<t}^i) \sum_{æ_{\leq T}^j,\ j\neq i} \prod_{t=1}^T \sigma(e_t|æ_{<t}a_t) \prod_{t=1}^T \prod_{j\neq i} \pi_j(a_t^j|æ_{<t}^j)}{\prod_{t=1}^T \pi_i(a_t^i|æ_{<t}^i) \sum_{æ_{<T}^j a_T^j,\ j\neq i} \prod_{t=1}^{T-1} \sigma(e_t|æ_{<t}a_t) \prod_{t=1}^T \prod_{j\neq i} \pi_j(a_t^j|æ_{<t}^j)}$$

$$= \frac{\sum_{æ^j_{\leq T},\ j \neq i} \prod_{t=1}^{T} \sigma(e_t | æ_{<t} a_t) \prod_{t=1}^{T} \prod_{j \neq i} \pi_j(a_t^j | æ^j_{<t})}{\sum_{æ^j_{<T} a^j_T,\ j \neq i} \prod_{t=1}^{T-1} \sigma(e_t | æ_{<t} a_t) \prod_{t=1}^{T} \prod_{j \neq i} \pi_j(a_t^j | æ^j_{<t})}$$

As desired, all appearances of π_i cancel.