Evolutionary Prediction Games

Jeffrey A. Barrett

Michael Dickson

Gordon Purves

Abstract. We consider an extension of signaling games to the case of prediction, where one agent ('sender') perceives the current state of the world and sends a signal. The second agent ('receiver') perceives this signal, and makes a prediction about the next state of the world (which evolves according to stochastic but not entirely random 'laws'). We suggest that such games may be the basis of a model for the evolution of successful theorizing about the world.

1. Introduction.

Recent work on signaling games has demonstrated that they can be used to model a number of phenomena, far beyond Lewis' (1969) original target of the successful communication via conventional meaning. This paper explores a closely related type of game, prediction games, as a rudimentary model of the emergence of successful theorizing about the world in terms of a similarly conventional signaling system.

In a simple iterated signaling game there are N states (sometimes referred to as states of the world, sometimes as states of the first player), chosen randomly (typically from a stationary, often flat, distribution) in each round of the game. The first player (the 'sender') produces a 'signal' chosen from a set of S many potential signals according to a distribution determined by the state. The second player (the 'receiver') then performs an 'act' chosen from a set of A many potential acts according to a distribution determined by the signal. (In other words, receiver 'knows' the signal, but not the state.) We say that signals and acts are determined by 'dispositions', which are sets of N many (for sender) or

S many (for receiver) probability distributions over the signals (for sender) or acts (for receiver). In the games we considered, these distributions are initially flat. There is a utility measure over states, signals, and acts for both sender and receiver.

The games that we consider here are cooperative—maximizing expected utility for one player entails maximizing it for both. Indeed, the games we consider have a stronger property, namely, that the payoffs to sender and receiver are always the same. (This condition could be weakened considerably, however, with no change in the analysis.)

Finally, sender and receiver are given the capacity to learn, in the following sense. After each iteration, if the payoff to sender and receiver is positive, then both receiver and sender are 'reinforced', meaning that their distributions are modified so that they are more likely to repeat their respective actions (signal for sender, act for receiver) under the same circumstances (state of the world for sender, signal for receiver). We discuss the nature of this 'learning algorithm' in detail below. It is sufficient for now to note that under a reasonable choice for this algorithm, in the simplest case ($N = S = A = 2$), both sender and receiver evolve (with probability 1) a completely efficient signaling system, which is to say that their distributions are given by one of the two signaling systems in Figure 1. In that figure (and henceforth) the rows of the matrices are the distributions for the various inputs. Hence, for example, in 1(a), if the state of the world is 1 (where we simply label the states $1, 2, \ldots, N$, and likewise for the signals and acts), then the first row of sender's matrix is used to determine a signal—i.e., sender will produce signal 1—and then, of course, receiver will perform act 1.

Sender	Receiver	Sender	Receiver
$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
(a)		(b)	

Figure 1. Signaling systems for $N = S = A = 2$.

Although we use suggestive terms such as 'sender', 'signal', and so on, they are to be taken as defined entirely by the formal rules governing their behavior. In other words, for example, there is no (presumed) sense in which 'senders' *understand* what they are doing when they 'send a signal', nor in which they 'intend to communicate'. The entire process is mechanical and could be redescribed in terms of stimuli and responses. We sometimes use less mechanistic terminology only for ease of exposition.

A learning algorithm that (provably) produces the kind of signaling system noted above in the case $N = S = A = 2$ is simple reinforcement learning, in which each agent has a matrix of 'propensities' (each represented as a positive integer), each initially set to the same value. When the pair is successful, the size of the propensity corresponding to the performed action (the input and output) is increased by 1 (or some other fixed number). Probabilities are determined from these propensities in the obvious way (divide the propensity by the sum of all propensities in its row). This algorithm is familiar as the 'basic model' used by Roth and Erev (1995), who discuss its pedigree and relation to evolutionary dynamics.

In the remainder of this paper, we will explore some modifications of this basic game, leading up to the 'prediction game', where the pair succeeds only if the receiver's act matches the *next* state of the world. (Of course, the state will need to evolve in some law-like fashion for there to be any hope of more than purely random success.) In section 2 we discuss some relevant extensions and modifications of the standard signaling game described above. We then (section 3) discuss prediction games, and present some results from computer simulations. The results support the very general idea that emergence of the ability to theorize about the world may be modeled by a prediction game, and we discuss this point in section 4.

2. Extensions of Signaling Games.

It is not difficult to imagine myriad extensions and modifications of the basic signaling game sketched above. Here we consider just those that will show up in our present study of prediction games.

2.1. Generalizations.

First, there are a number of points where the model may be generalized. One could allow that $N \neq S \neq A$. Moreover, one could allow that senders do not react to all world-states, i.e., there are more world-states than the sender can 'see', or that receivers do not react to all signals, i.e., the sender has more signals than the receiver can 'hear'. (Denote the number of states that the sender can discern by $\tilde{N}$ and the number of signals that the receiver can discern by $\tilde{S}$.) As we will have occasion to notice in a moment, these modifications have surprisingly little impact on the basic result noted above. Of course, if the pair does not have sufficient resources to develop a *fully* successful signaling system, then they will not do so, but in general they may (and often do) come as close as they logically can. For example, with the basic learning model described above and with $N = 4$, $S = 3$, and $A = 3$, a pair whose sender can see all four world-states ($\tilde{N} = 4$) and whose receiver can see all three signals ($\tilde{S} = 3$) is likely to evolve dispositions such as those in Figure 2, where each number in the matrices is a probability (derived, of course, from the propensities).

$$\text{Sender} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1/3 & 1/3 & 1/3 \end{pmatrix} \qquad \text{Receiver} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Figure 2. Evolved dispositions in the case $N = 4$, $S = 3$, $A = 3$, $\tilde{N} = 4$, $\tilde{S} = 3$.

This pair will never get things right when the world-state is 4 (the receiver cannot perform act 4), but are nonetheless as successful as they could be given the situation. (In section 2.3 we will develop a few ideas to make such statements more precise. The numbers in Figure 2 are made up only to illustrate a point. We will discuss the actual results of actual simulations later.)

Senders and receivers might not be completely accurate in what they see and hear—instead of seeing the world-state every time, a sender might occasionally see some 'nearby' state. Below, we implement this idea by assigning to senders a 'state perception standard deviation', Δ_σ, so that the state that actually serves as input to the sender's disposition is chosen according to a normal distribution centered at the actual world-state, with standard deviation Δ_σ (then rounded to the nearest whole number). For this and similar determinations, the state space has the topology of a ring. A similar idea can be applied to the receiver by choosing a 'signal perception standard deviation', Δ_ρ.

As we noted above, there are numerous other modifications one might wish to consider. One could allow senders and receivers to 'change partners' from time to time, perhaps as they move around on a grid. Or perhaps receivers can hear more than one sender at a time. Perhaps one and the same agent could be both sender and receiver. And so on. Many of these (and other) extensions are also worthy of study in the context of prediction games, but here we restrict ourselves to those already noted, in large part to demonstrate the robustness of the results under the introduction of these complications.

2.2. Learning Algorithms.

Let sender's propensity on round t of the game, for output s on input n, be a positive integer, denoted $\sigma_{ns}(t)$. Similarly, receiver's propensity on round t of the game for output a on input s is a positive integer, $\rho_{sa}(t)$. ($\sigma(t)$ and $\rho(t)$ are thus matrices, examples of which we saw above.) We always presume that for some integer, B, $\sigma_{ns}(0) = \rho_{sa}(0) = B$ for all n,

s, and a. The basic learning algorithm mentioned above says that

$$\begin{aligned} \sigma_{ns}(t+1) &= \begin{cases} \sigma_{ns}(t) - l & \text{on failure} \\ \sigma_{ns}(t) + k & \text{on success} \end{cases} \\ \rho_{ns}(t+1) &= \begin{cases} \rho_{ns}(t) - l & \text{on failure} \\ \rho_{ns}(t) + k & \text{on success} \end{cases} \end{aligned} \tag{1}$$

for some positive integer, k, and $l = 0$. As we said above, probabilities are derived from these propensities by

$$\begin{aligned} \Pr_\sigma(s|n)(t) &= \frac{\sigma_{ns}(t)}{\sum_i \sigma_{ni}(t)} \\ \Pr_\rho(a|s)(t) &= \frac{\rho_{sa}(t)}{\sum_i \rho_{si}(t)}. \end{aligned} \tag{2}$$

In addition to the 'reward' k, one might wish to impose a 'penalty' for failure, which we do by choosing $l > 0$. With penalties, it is possible for a propensity to become negative, which would of course pose a problem for (2). In general, then, we specify a minimum size for any propensity, which we will always set to 1. Another reasonable choice is 0. However, the latter choice does preclude the possibility of 'unlearning' an unsuccessful strategy.[1] Learning with both rewards and penalties tends to be slightly more successful (in the precise senses discussed below) than it is with only rewards.

In addition to adding penalties to the basic learning algorithm, we consider two further dynamical modifications to agents' dispositions, very similar to what Roth and Erev (1995) call 'persistent local experimentation' and 'gradual forgetting'.[2]

In our models, local experimentation is implemented by associating standard deviations, x_σ and x_ρ, with senders and receivers, respectively. An agent first chooses a provisional output according to the relevant distribution, but the *actual* output is then

[1] In actual simulations, with reasonable choices for the other parameters, the difference between these two choices is small but noticeable.

[2] They also introduce 'extinction in finite time', which forces probabilities to go to zero below some cutoff. We do not consider this modification here, in part because it is often effectively nullified by local experimentation in many of the cases we care about.

chosen according to a normal distribution centered on the provisionally chosen output, with the given standard deviation. As before, we round the choice to the nearest whole number and mod by the number of outputs.[3]

Gradual forgetting allows one to maintain the overall magnitude of the propensities at a reasonable maximum in a natural way.[4] Doing so ensures that even after a long time, present success or failure effects dispositions about the same as medium-term successes or failures did. (On the other hand, early successes and failures still have a much larger influence on dispositions. This situation is desirable, because it means that the 'learning curve' is very steep in the early stages of the game, allowing agents very quickly to evolve successful dispositions.[5]) We implement forgetting by multiplying each propensity by a factor, ϕ, every F rounds of the game.[6] Of course, $0 < \phi \leq 1$, with equality on the right if there is no forgetting. Forgetfulness can improve success by giving agents who early on made some bad choices a chance to unlearn what they think they know. Without forgetfulness, the size of the propensities can grow so large that before long, even a long string of failures cannot affect the agent's disposition appreciably. (The fact that agents tend to evolve decent signaling systems even with $\phi = 1$ is a testament to the rarity of evolving 'badly' in the early stages.)

2.3. Measures of Success.

What do we mean by 'success'? There are (at least) three questions that we might ask about sender-receiver pairs evolving according to the dynamics given above.

[3]Our implementation is thus not quite that of Roth and Erev. In their model, 'experimentation' occurs only when the provisional output has high probability, and then only 'nearby' outputs have a chance of becoming the true ('experimental') output. In our model, experimentation is always a possibility and all outputs have non-zero chance of becoming the experimental output (though for small standard deviations, only nearby outputs have an appreciable chance of being selected). These differences do not appear to lead to substantially different (or even noticeably different) behavior, for reasonable choices of x_σ and x_ρ.

[4]Keeping the magnitude down has an additional benefit: the probabilities never get so small that one needs to worry about rounding errors due to limits in the precision of the computer.

[5]In fact, the situation is more complicated. *Very* early penalties and rewards *also* tend not to have a big influence on agents' dispositions. See note 9 for why.

[6]In Roth and Erev's models, $F = 1$. Allowing $F > 1$ and adjusting ϕ accordingly yields nearly identical results, and allows simulations to run faster.

First: How quickly (if ever) do they become 'stable', i.e., their dispositions do not change much over time? To address this question, we need a measure of the 'difference' between the agent's disposition before and after it is updated in response to success or failure. There is more than one way to skin that cat, but for our purposes the differences among them do not appear to be significant. We take the difference between two dispositions to be the *mean* Kullback-Liebler (KL) divergence (the relative entropy) between the rows of the dispositions. For two dispositions, $\Pr(i|j)$ and $\Pr'(i|j)$ (i and j are indices to the outputs and inputs of the disposition respectively), which we can conceive as distinct sets (indexed by j) of conditional probability measures, the 'difference' between them is thus

$$\mathbf{d}[\Pr(i|j), \Pr'(i|j)] = \sum_j \sum_i \left[\frac{1}{J} \Pr(i|j) \log_2 \left(\frac{\Pr(i|j)}{\Pr'(i|j)} \right) \right], \tag{3}$$

where J is the number of rows in the dispositions.[7] Finally, by choosing some threshold, τ, we can say that a disposition is 'stable' if the KL-divergence between the disposition at t and $t+1$ remains below this threshold for 'sufficiently long' (another number, T, that we must choose). Figure 3 may give the reader some sense of the numerics.

$$\Pr_1 \quad \begin{pmatrix} 0.97 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.97 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.97 \end{pmatrix} \qquad \Pr_2 \quad \begin{pmatrix} 0.96 & 0.02 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.97 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.97 \end{pmatrix} \qquad \Pr_3 \quad \begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}$$

Figure 3. Mean KL-divergence of two pairs of 4x4 dispositions. $\mathbf{d}(\Pr_1, \Pr_2) \approx 0.0011$ (the only difference between them appears in the first row), and $\mathbf{d}(\Pr_1, \Pr_3) \approx 1.7581$.

In practice, it is often too time-consuming to calculate the KL-divergence between the dispositions at t and $t+1$ and instead we choose some positive integer, Π—the 'period'—and calculate the KL-divergence between the dispositions sampled at intervals Π, saying that an agent's disposition is 'stable' if this KL-divergence is below τ for T periods

[7]The KL-divergence is not a metric on probability measures. Nonetheless, it is a convenient and simple 'measure' of how close two probability measures are, and the functional in (3) gives one a pretty good handle on how 'stable' a disposition is. I.e., if it is small for a while, then the disposition is stable.

(where T is of course positive). A similar procedure will be used to check on the evolution of the other two measures, mentioned below. This procedure can hide very short term (occurring over intervals less than Π) fluctuations, but longer-term trends (which are our main concern here) are faithfully captured by these measures.[8] Note that an agent may be 'stable' at one stage of the simulation, and 'unstable' at another—its disposition could be stable for a while, and then become unstable, either randomly or in response to some changes in the conditions of the simulation.[9]

Our next question concerns accuracy: How well do a pair's dispositions get the receiver's act to match the state of the world? And how does this accuracy evolve as the agents' dispositions evolve? It is a simple matter to calculate the overall probability that a pair will be successful on any given round of the game, given their dispositions:

$$\sum_{n \in \{1 \ldots N\}} \sum_{s \in \{1 \ldots S\}} \frac{1}{N} \mathrm{Pr}_{\rho}(n|s) \mathrm{Pr}_{\sigma}(s|n). \tag{4}$$

(Note that if the world-states are not chosen from a flat distribution then we would replace the factor $1/N$ with their distribution.[10]) In the end, we are interested in the *mean* accuracy of all agents, i.e., we take the mean of (4) over all pairs. We report accuracy as a probability, expressed as a percentage between 0 and 100.

Third, how 'efficient' is a given pair? One natural understanding of 'efficiency' is that a pair is efficient if and only if it does the best it can to associate, deterministically, to each act of the receiver a state of the world. Consider, as an example of *inefficiency*, the

[8] A more fine-grained notion of stability can still, of course, be pursued by setting $\Pi = 1$, and in this case, one finds that once a disposition hits long-term stability, the short-term fluctuations are indeed generally small.

[9] When the number of states, N, is large, and the initial size (B) of the propensities is large relative to k and l, agents might, by our criteria, become 'artificially' stable in the early stages of the simulation, because the denominator in (2) is very large relative to the size of the individual propensities, which themselves change a relatively small amount per reward or penalty. After a while, however, the dispositions typically become unstable, before much later becoming stable again.

[10] In the dynamical case that we consider below, the world-states are indeed not chosen from a flat distribution, but instead according to a dynamical law. The condition in that case is that over long stretches of time, the states appear with equal frequency. This condition holds in all of the dynamical laws that we consider.

pair of distributions in Figure 4.

$$\text{Sender} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{Receiver} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1/3 & 1/3 & 1/3 & 1/3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Figure 4. An inefficiency pair of dispositions.

It represents a commonly encountered inefficiency (called 'partial pooling equilibrium') in which the pair is inefficient because it fails to associate any state to the act 3. Note that the 'blame' here appears to fall squarely on the sender. The receiver is 'doing the best it can' given the sender's refusal to use signal 3. The receiver 'hears' 2 when the world-state is both 2 and 3, and the best pure strategy that the receiver can evolve is to choose one or the other and always do that act in response to signal 2. The receiver could also have chosen a mixed strategy on input 2 (sometimes choosing act 2, sometimes act 3) without loss of success (accuracy). Contrast this pair with the one in Figure 2. The latter is efficient (by our definition), despite the persistent failure on state 4. Given their limited resources, they are doing 'as well as they can'—they are not 'wasting' any signals.

Rather than simply classifying pairs as efficient or not, it is helpful (especially in situations where full efficiency is unlikely, such as when N, S, and A are large) to take a more fine-grained approach. Let the *fine-grained efficiency* for a pair be the *number* of acts that are deterministically associated to states, divided by the total number that *could* be, by this pair, given the parameters N, S, A, $\tilde{N}$, and $\tilde{S}$.[11] This measure is especially pertinent in cases where full efficiency is very unlikely.

In practice, because we do not allow propensities to go to zero (recall that the

[11]This measure is not perfect. Consider the pair from Figure 4. Their fine-grained efficiency is 75%, but if the receiver had played a mixed strategy on state 2—arguably just as efficient a use of resources—their efficiency would have been 50%. However, the complexity of implementing a better definition of fine-grained efficiency is not worth the small gain—in the situations that we consider, agents very rarely end up playing mixed strategies when they can play a pure strategy, and we are only after a very general picture of overall efficiency. We can just assume that the actual mean fine-grained efficiency (i.e., properly accounting for mixed strategies) across a large group of agents is ever so slightly larger than what we report here.

minimum size of any propensity is 1), we cannot require complete determinism for efficiency. Instead, we choose some threshold, ϵ, and require that the probability be greater than $1 - \epsilon$. The statistic of interest is the percentage of pairs that evolve efficient signaling systems (and thus avoid unnecessary partial pooling equilibria), or in some cases (especially when partial pooling is extremely common) the mean fine-grained efficiency.

2.4. Examples.

We now consider some specific examples and relate the results of their simulation,[12] for later comparison with prediction games. In these examples, we declare an agent to be 'stable' if the mean KL-divergence between its current disposition and its previous disposition is below $\tau = 0.0025$ for $T = 10$ periods of $\Pi = 100$ rounds each, and we report, for each simulation, the percentage of agents that are stable by the end. 'Accuracy' is the mean over all pairs in the simulation. Efficiency is fine-grained efficiency. Each result reported below is the result of allowing the game to run until the three quantities of interest have 'settled down'.[13]

We report the results of a simple game in Table 1.[14] Stab1 refers to the mean time

[12]Simulations were done in the NetLogo platform (http://ccl.northwestern.edu/netlogo/), with two of us independently coding the simulations and comparing results. Simulations are run on multiple pairs only to save computing time—in these simulations, the pairs 'know nothing' of one another (though they do see the same random sequence of world-states, so it is important to run the simulations multiple times to confirm the general results).

[13]The fine print: Mean accuracy, in every case we have seen, settles down to a near constant value well before efficiency and stability do. (See Figure 5 for an example). Depending on the conditions of the simulation, efficiency and especially stability can continue to experience fluctuations into the long term, but after a while, these fluctuations occur around a near-constant mean. In order to detect when to halt the simulation, we choose a number, I, of periods over which to time-average the quantities. (Recall that these quantities are 'sampled' once per every Π rounds, where Π rounds is a 'period'). We then choose a number, ν, of consecutive time-averaged quantities to remember. We check the standard deviation of the most recently collected ν many time-averaged values, and if it is low enough for both stability and efficiency, we halt. In all but the most chaotic simulations, these standard deviations can be kept quite low (typically around 0.25 for efficiency, taken as a number between 0 and 100, and 0.5 for stability, also a number between 0 and 100, with $I = 10$ and $\nu = 5$) and still we get halting in reasonable time—tens of thousands of rounds for the cases with the lowest numbers of states and less randomness, up to hundreds of thousands of rounds for the cases with many states and more randomness. (The actual 'first moment' at which the quantities have settled is of course somewhat earlier. In the typical (for us) case of $\Pi = 100$, $I = 10$ and $\nu = 5$ it happened 5,000 rounds earlier.)

[14]We report stability, accuracy, and efficiency only to two significant figures, as re-running the simulations can lead to changes in these quantities that would make reporting a more precise figure misleading. For

to first stability, which gives a very general sense of how quickly the pairs become stably successful. Efficiency in reported results refers to fine-grained efficiency.[15] A few

	N	S	A	$\tilde{N}$	$\tilde{S}$	Δ_σ	Δ_ρ	x_σ	x_ρ	Stab.	Acc.	Eff.	Stab1
(i)	4	4	4	4	4	0	0	0	0	98%	96%	96%	4314

Table 1. Results of a signaling game simulation run on 10000 pairs. Learning dynamics: $B = 100$, $k = 2$, $l = 1$, $x_\sigma = x_\rho = 0$, $\phi = 0.096$, $F = 100$. Stability: $\tau = 0.001$, $T = 10$, $\pi = 100$. Efficiency: $\epsilon = 0.1$.

observations about these results apply as well to our other results, reported below.

First, these results are a 'snapshot in time' of what is happening in the simulation. Hence, for example, the 99% stability in (i) does not necessarily mean that a couple of agents have stubbornly refused to become stable, but more likely that, at the moment that this snapshot was taken, these agents (who were in all likelihood stable at some earlier moment) have experienced enough of a fluctuation in their dispositions to become temporarily unstable. (On the other hand, it is true that agents who have evolved to be inefficient tend to be more susceptible to fluctuations and therefore long-term instability.)

Efficiency is a different story. Typically, efficient agents do not become inefficient, and agents who have been inefficient for a long time do not tend to become efficient. Instead, they tend to evolve dispositions such as those in Figure 4.

Finally, it is worth bearing in mind that although simulation (i) was run to around

example, in 10 runs of simulation (i), the means and standard deviations for stability, accuracy, and efficiency were:

	Mean	Standard Deviation
Stability	*xxx*	*xxx*
Accuracy	*xxx*	*xxx*
Efficiency	*xxx*	*xxx*

The number of rounds required to reach very stable values for these quantities (see note 13 for a discussion of halting) ranged from 23,000 to 39,200.

[15]Note, as well, the seemingly large value for ϵ. For games with small numbers of inputs and outputs, one can set ϵ quite small (e.g., 0.01) and still see high rates of efficiency. However, with higher numbers of inputs and outputs, a minimum size for each propensity, and an effective cap on the size of any propensity (due to forgetting), even a successful pair 'doing the best it can', with no partial pooling, could easily have conditional probabilities only around 0.95, so that their product is already nearly 0.9, right at the threshold for being accounted efficient. Indeed, for the 16x16x16 game, frequently agents who appear to be doing quite well have poor efficiency even with $\epsilon = 0.1$, and it can be instructive in that case to set ϵ even higher.

24,000 rounds, in fact the fate of the vast majority of the pairs is sealed in the early stages. As early as several hundred rounds into the game, most agents have moved enough towards a 'favored' output for each input that turning back is very unlikely. Indeed, notice that the mean time of first stability in this simulation was 4283 rounds.[16] Not long after this time, accuracy begins to rise sharply. Changing various parameters in the learning model (such as reward and penalty) can change these numbers, but not the general shapes of these curves (except when extreme values are chosen). After the curves flatten out, very little happens thereafter apart from some tiny fluctuations. These fluctuations can be made greater, of course, by increasing forgetfulness (among other things).[17] Figure 5 shows of all three curves for simulation (i).

It is a simple matter to explore various changes in the parameters of the simulation. We illustrate some additional (and typical) results in Table 2. The basic fact that many pairs evolve a more or less successful (stable, accurate, and efficient) signaling system is remarkably robust under changes in the parameters. We leave it to the reader to observe

	N	S	A	$\tilde{N}$	$\tilde{S}$	Δ_σ	Δ_ρ	x_σ	x_ρ	Stab	Acc	Eff	Stab1
(i)	4	4	4	4	4	0	0	0	0	98%	96%	96%	4314
(ii)	4	4	4	4	4	0.25	0.25	0	0	95%	96%	96%	4511
(iii)	4	4	4	4	4	0.50	0.50	0	0	45%	85%	87%	7257
(iv)	4	4	4	4	4	0	0	0.25	0.25	96%	96%	96%	4557
(v)	4	4	4	4	4	0.25	0.25	0.25	0.25	94%	97%	96%	4913
(vi)	4	4	4	4	4	0.50	0.50	0.25	0.25	7%	94%	98%	18398
(vii)	4	3	4	4	3	0	0	0	0	95%	74%	97%	4108
(viii)	4	4	4	3	5	0	0	0	0	97%	75%	100%	4381
(ix)	8	8	8	8	8	0	0	0	0	xx%	xx%	xx%	xxxx
(x)	8	8	8	8	8	0	0	0.25	0.25	89%	93%	96%	7625

Table 2. Results of several signaling game simulations run on 10000 pairs. Learning dynamics: $B = 100$, $k = 2$, $l = 1$, $\phi = 0.096$, $F = 100$. Stability: $\tau = 0.0025$, $T = 10$, $\pi = 100$. Efficiency: $\epsilon = 0.1$.

[16] Recall that agents can be stable then become unstable. The mean *last* time of stability was 5675 in this simulation. Also, the 'first time' of stability does not include any 'artificial' stability from very early in the simulation—see note 9.

[17] Recall that a higher forgetfulness will lower the maximum size of a disposition, so that smaller absolute changes in size have a larger effect. Hence stability, but also efficiency and accuracy, can suffer if the maximum size is too small, i.e., forgetfulness is too high (ϕ is small or F is small or both).

the various trends in the results. Our main point, for now, is the one already made—the evolution of signaling systems is robust.

3. Prediction Games.

3.1. The Model.

A natural extension of signaling games is to allow the state of the world to evolve according to some simple laws, and then require that the receiver 'predict' the next state of the world, in the sense that success for the pair requires the receiver's act at t to match the state of the world at $t+1$, reward or penalty being applied at $t+1$. One might be inclined to choose, for these laws, some 1-1 map from the state space for the world to itself (a permutation of the space). However, in that deterministic case, and because the meaning of signals is purely conventional, predicting the future is tantamount to knowing the present. The results of simulations carried out in such a deterministic case are exactly as they are in the cases described above, so long as the laws guarantee that each state is visited with roughly equal frequency. A simple example of such a law is one where the state is incremented by 1 each round. We say that there is a 'drift' of 1 in the laws. (As before, the state space is taken to be a ring for this purpose.) Hence everything that one has learned about signaling games can be reconstrued in terms of a deterministic prediction game in which signals come to refer to the future (rather than the present) state of the world.

To get genuinely new behavior, we need the world to be slightly unpredictable. A simple way to achieve the required randomness is to introduce, in addition to a drift, y, a standard deviation, z. Let the state at t be $n(t)$. Then the state at $t+1$ is chosen from a normal distribution, centered at $n(t)+y$, with a standard deviation z. So the world *tends* to drift by an amount y each turn, but occasionally (or frequently if z is allowed to be large) it does something else. For the values of z typically chosen in our simulations, that 'something else' is nearly always to drift an additional unit, or not to drift at all. Additional randomness can be introduced by allowing for an occasional 'surprise'—on each round, with probability β, the state is chosen at random from a flat distribution. (Other choices for this distribution could of course be made.)

As before, one may be interested in a pair's ability to evolve dispositions that are as successful as possible (or nearly so). In the indeterministic case, however, there are no dispositions that could achieve 100% success. Instead, we account a pair as 'making the right prediction' if it predicts the drifted state.[18] Hence, while we continue to use the term 'accuracy', it is a slight misnomer in this case, referring to how likely the pair is to make the best (but not necessarily correct) prediction.

3.2. Results.

For easy comparison, we first show, in Table 3 results for an indeterministic version of simulations (i)-(x). A few observations are immediately apparent. First and foremost, the

	N	S	A	$\tilde{N}$	$\tilde{S}$	Δ_σ	Δ_ρ	x_σ	x_ρ	Stab	Acc	Eff	Stab1
(i′)	4	4	4	4	4	0	0	0	0	87%	94%	95%	5177
(ii′)	4	4	4	4	4	0.25	0.25	0	0	80%	93%	94%	5508
(iii′)	4	4	4	4	4	0.50	0.50	0	0	1%	63%	58%	17421
(iv′)	4	4	4	4	4	0	0	0.25	0.25	74%	96%	97%	6437
(v′)	4	4	4	4	4	0.25	0.25	0.25	0.25	61%	95%	97%	7105
(vi′)	4	4	4	4	4	0.50	0.50	0.25	0.25	0%	89%	89%	73067
(vii′)	4	3	4	4	3	0	0	0	0	51%	73%	97%	6097
(viii′)	4	4	4	3	5	0	0	0	0	91%	74%	100%	5281
(ix′)	8	8	8	8	8	0	0	0	0	xx%	xx%	xx%	xxxx
(x′)	8	8	8	8	8	0	0	0.25	0.25	51%	89%	95%	19794

Table 3. Results of a single signaling game simulation run on 10000 pairs. World Laws: $y = 1$, $z = 0.5$, $\beta = 0$. Learning dynamics: $B = 100$, $k = 2$, $l = 1$, $\phi = 0.096$, $F = 100$. Stability: $\tau = 0.001$, $T = 10$, $\pi = 100$. Efficiency: $\epsilon = 0.1$.

pairs do, quite robustly, evolve successful prediction systems, even in cases where conditions make stability very difficult to achieve.[19]

[18] For pathologically large values of z or β, while predicting the drifted state is still the best one can do, it will in fact be fairly unsuccessful, because the world is essentially random. Of course, agents are not able to learn much in such a world (there is not much to learn), and tend not to evolve any stable prediction system, successful or otherwise.

[19] One trend not shown in the table is that the accuracy of converged pairs is—somewhat unsurprisingly—higher than average. For example, the accuracy of just the converged pairs in both (i′) and (ii′) is 99% while

The main difference between the signaling and prediction games is in the sensitivity to added randomness (Δ_σ, Δ_ρ, x_σ, x_ρ)—these factors have a larger effect on convergence especially, and to a lesser extent, accuracy and efficiency. One other interesting difference is explored in the next section.

3.3. Learning New Tricks.

The indeterministic case introduces another type of learning that has no correlate in the deterministic case. Suppose that a pair has evolved a reasonably accurate prediction system. Now, for whatever reason, the laws change. (For example, perhaps they learned to predict in one environment, but now find themselves in another.) Can the pair adapt? Can they 'unlearn' their old ways and learn the new laws? In such cases, who makes the adjustment? In principle, of course, one of them (either sender or receiver) could stubbornly refuse to change its disposition, forcing the other to change. But what actually happens? We turn now to some results.

Our procedure here is to run the simulations to 'completion', as above, then, keeping the agents as they are, change the laws (by changing the drift term, y, from $y = 1$ to $y = 3$), and the continue the simulation. Of course, the pairs are initially horrible at predicting under the new circumstances, but, as the results in Table **??** show, they pretty quickly develop successful prediction systems. In Table **??**, Stab1 now refers to the first time at which the pairs converge on their *new* system—it is therefore a measure of how quickly they adapt to the new situation.

4. Theories as Prediction Systems.

Faced with the task of modeling the evolution of the ability to predict, a likely first attempt would be to model a single agent who witnesses a state of the world and acts. The agent would be rewarded as above for a successful prediction, and penalized for an

for (iii′) it is 94%. Similar figures hold for the other cases, apart, of course, from (vi′).

	N	S	A	$\tilde{N}$	$\tilde{S}$	Δ_σ	Δ_ρ	x_σ	x_ρ	Stab	Acc	Eff	Stab1
(i′)	4	4	4	4	4	0	0	0	0	87%	94%	95%	5177
(ii′)	4	4	4	4	4	0.25	0.25	0	0	80%	93%	94%	5508
(iii′)	4	4	4	4	4	0.50	0.50	0	0	1%	63%	58%	17421
(iv′)	4	4	4	4	4	0	0	0.25	0.25	74%	96%	97%	6437
(v′)	4	4	4	4	4	0.25	0.25	0.25	0.25	61%	95%	97%	7105
(vi′)	4	4	4	4	4	0.50	0.50	0.25	0.25	0%	89%	89%	73067
(vii′)	4	3	4	4	3	0	0	0	0	51%	73%	97%	6097
(viii′)	4	4	4	3	5	0	0	0	0	91%	74%	100%	5281
(ix′)	8	8	8	8	8	0	0	0	0	xx%	xx%	xx%	xxxx
(x′)	8	8	8	8	8	0	0	0.25	0.25	51%	89%	95%	19794

Table 4. Results of a single signaling game simulation run on 10000 pairs. World Laws: $y = 1$, $z = 0.5$, $\beta = 0$. Learning dynamics: $B = 100$, $k = 2$, $l = 1$, $\phi = 0.096$, $F = 100$. Stability: $\tau = 0.001$, $T = 10$, $\pi = 100$. Efficiency: $\epsilon = 0.1$.

unsuccessful prediction. It will come as no surprise that such an agent will evolve a successful predictive disposition. Indeed, rates of success are *higher* than they are for our prediction systems. (This result is also unsurprising—such an agent does not face the additional task of ‘communicating’ along with predicting.)

However, in our view, while a model such as this one may be sufficient for capturing how a mechanistically described agent may ‘learn’ to predict the state of the world, it fails miserably as an even rudimentary model for the evolution of the capacity to *theorize* about the world, for a few reasons. Our model, on the other hand, is a valid first-step, and has the capacity to be extended in ways (some of which we describe below) that will capture some subtler aspects of theorizing.

Basic theme: Bona fide theorizing about the world involves representation, whereas a single ‘learner’ who simply changes behavior in response to input states does not have a representation of the world. Signaling/predicting pairs do—the signal is the ’representation’. (Note that the ‘pair’ could in fact be aspects of a single agent, a predictor. This predictor gets an input, and them by internal machinations modeled by the sending and receiving of a signal, eventually issues a ‘prediction’. Those internal machinations are what amount to the predictor’s ‘representation’ of the world.)

OK, a bit of fleshing out is obviously needed here too... One obvious thing to at least mention is the potential for combining Jeff's coding stuff with this prediction stuff. Then we start to see a more interesting notion of a 'theoretical language' develop and stuff that Jeff has said about partitioning and incommensurability could be applied in the context of prediction (and thus, theorizing in a richer sense). I would like to pursue that as a next step (not in this paper.)

sim_i_graphs.png

Figure 5. Accuracy, efficiency (fine-grained), and stability in simulation (i). The red curve (the last to rise from zero) is stability. The green curve (the one that begins at 24%) is accuracy, and the black curve (first to rise from zero) is fine-grained efficiency.

REFERENCES

Barrett, J. A. (2007), "The Evolution of Coding in Signaling Games", *Theory and Decision* **xx**:xx–xx.

Barrett, J. A. (2008), "Dynamic Partitioning and the Conventionality of Kinds", *Philosophy of Science* **74**:527–546.

Lewis, D. (1969), *Convention: A Philosophical Study*. Harvard University Press: Cambridge, MA.

Roth, A., and I. Erev (1995), "Learning in Extensive-Form Games: Experimental Data and Simple Dynamics Models in the Intermediate Term", *Games and Economic Behavior* **8**:164–212.

Skyrms, B. (2009), *Signals: Evolution, Learning, and the Flow of Information*. xx:xx.