Numerical Simulations of the Lewis Signaling Game: Learning Strategies, Pooling Equilibria, and the Evolution of Grammar

Jeffrey A. Barrett
22 September 2006

Abstract

David Lewis (1969) introduced sender-receiver games as a way of investigating how meaningful language might evolve from initially random signals. In this report I investigate the conditions under which Lewis signaling games evolve to perfect signaling systems under various learning dynamics. While the 2-state/2-term Lewis signaling game with basic urn learning always approaches a signaling system, I will show that with more than two states suboptimal pooling equilibria can evolve. Inhomogeneous state distributions increase the likelihood of pooling equilibria, but learning strategies with negative reinforcement or certain sorts of mutation can decrease the likelihood of, and even eliminate, pooling equilibria. Both Moran and APR learning strategies (Bereby-Meyer and Erev 1998) are shown to promote successful convergence to signaling systems. A model is presented that illustrates how a language that codes state-act pairs in an order-based grammar might evolve in the context of a Lewis signaling game. The terms, grammar, and the corresponding partitions of the state space co-evolve to generate a language whose structure appears to reflect canonical natural kinds. The evolution of these apparent natural kinds, however, is entirely in service of the rewards that accompany successful distinctions between the sender and receiver. Any metaphysics grounded on the structure of a natural language that evolved in this way would track arbitrary, but pragmatically useful distinctions.

I. Introduction to Lewis Signaling Games

A Lewis (1969) signaling game has two players: the sender and the receiver. On an *n*-state/*n*-term Lewis signaling game there are *n* possible states of the world, *n* possible terms the sender might use as signals, and *n* possible receiver actions, each of which corresponds to a state of the world. If the action matches the state, then each player is rewarded.

Nature chooses a state at random on each play of the game. The sender then observes the state and sends a term to the receiver, who observes the signal but not the state. The receiver chooses an act based on the term he receives. On the basic game, there is exactly one act for each state that provides a positive payoff to both the sender and the receiver; other possible acts provide a payoff of zero. If the receiver's act matches the state, both players get the positive payoff, then a new state of is selected by nature, and the game is repeated.

The sender and receiver may learn from their experiences on repeated plays. Whether and how quickly they learn will depend on the learning strategy they use. If the sender and the receiver evolve to a state where they are more successful then chance, then they have evolved a more or less efficient language. The efficiency of the evolved language may be measured by the mean Shannon information content of each signal. Lewis called a system that evolves to a maximally efficient language a *signaling system*. For a signaling system in the basic Lewis signaling game, each state of the world corresponds to a term that corresponds to an act that matches the state of the world, and each signal leads to a successful action.

Brian Skyrms (2004, 2006) has studied Lewis signaling games extensively. Significant work has also been done by Simon Huttegger (forthcoming A and B). Among his results, Skyrms has shown that signaling systems always evolve in the 2-state/2-term Lewis signaling game with urn learning. This result suggests the possibility that natural languages may have developed from random signaling in a roughly analogous way. As we will see, however, prefect signaling systems sometimes fail to evolve even in basic Lewis signaling games with urn learning when there is more than two states and two acts. Given this, it is natural to ask for the conditions under which perfect signaling systems will evolve in Lewis signaling games.

The purpose of this report is investigate the conditions under which signaling systems evolve in simulations of Lewis signaling games. Several variations of the basic Lewis signaling game and a variety of alternative learning dynamics are considered. It is also shown how a multi-term grammar might evolve in the context of a Lewis signaling game. The results of the simulations reveal at lest some of the features to the evolution of linguistic convention in Lewis signaling games. As one might expect, the particular learning strategies adopted by the players are of central significance, but convergence to a signaling system is also contingent on more subtle features of the model.

The stochastic simulations discussed in this paper allow one to get a sense of the limiting behavior of wide variety of signaling systems, but (1) the apparent convergence to a perfect signal success rate does not guarantee convergence to a signaling system and (2) failure to reach a specified standard of success does not entail failure of convergence to a signaling system--in each case, additional arguments are needed to determine the long-term behavior of a system. The purpose of this report is primarily to present data from the numerical simulations, but auxiliary arguments concerning the limiting behavior of the systems described will be discussed as appropriate.

II. Basic Urn Models and Pooling Equilibria

In the basic 2-state/2-term Lewis signaling game with urn learning, there are two possible states of the world (*A* and *B*), two possible terms (*1* and *2*), and two possible acts (*A* and *B*) corresponding to the states. The sender has an urn labeled *state A* and an urn labeled *state B*, and the receiver has an urn labeled *signal 1* and an urn labeled *signal 2*. The sender's urns each begin with one ball labeled *signal 1* and one ball labeled *signal 2*, and the receiver's urns each begin with one ball labeled *act A* and one ball labeled *act B*.

On each play of the basic game, the state of the world is randomly determined with uniform probabilities, then the sender consults the urn corresponding to the current state and draws a ball at random, where each ball has the same probability of being drawn. The signal on the drawn ball is sent to the receiver. The receiver then consults the receiver urn corresponding to the signal and draws a ball at random. If the action on the drawn ball matches the current state of the world, then the sender and the receiver each return their drawn ball to the respective urn and add another ball to the urn with the same label as the drawn ball; otherwise, the sender and receiver just return their drawn ball to the respective urn. On the basic urn learning strategy, there is no penalty to the agents for the act failing to match the state. The game is repeated with a new state of the world.

Adding balls to the urns on successful acts changes the relative proportion of balls in each urn, which changes the conditional probabilities of the sender's signaling (conditional on the state) and the receiver's acts (conditional on the signal). The change in the proportion of balls of each type in each urn increase the likelihood that the sender and receiver will draw a type of ball that will lead to successful action.

In agreement with Skyrms' results, simulations of the 2-state/2-term basic urn model were observed always to approach to a signaling system. After 10^6 plays the ratio of successful actions to the number of plays, the signal success rate, is typically better than 0.999. The evolution to successful signaling systems is also uniform across independent runs. Out of 10^3 runs, there were three where the success rate was less than 0.8 after 10^6 plays, and each of these exceptional cases also eventually approached a signaling system.

This uniform convergence to signaling systems does not, however, hold for *n*-state/*n*-term Lewis signaling games more generally. If *n* is greater than two, then pooling equilibria may develop that prevent convergence to a perfect signaling system.

The following table shows the run failure rates for various Lewis signaling systems with more than two states, terms, and actions. There are 10^6 plays/run here, and a run is taken to fail to converge if the final signal success rate of the run is less than 0.8.

Model	**Run Failure Rate**
3-state/3-term	0.096
4-state/4-term	0.219
8-state/8-term	0.594

While failures to approach signaling systems were observed, each system always learned to do better than chance in signaling, and hence might be said to have developed a more or less effective language. In those cases where a perfect signaling system failed to evolve in the 3-state/3-term model, the system approached a signaling success rate of about 2/3. Since on chance alone one would expect a signal success rate of about 1/3, even those systems that fail to approach a perfectly efficient language nevertheless are always observed to learn a language. Systems that approached a signaling success rate of 2/3 here do not, as one might imagine, learn to signal reliably with two out of three terms; rather such systems approach a state where two of the signal terms correspond to the same state-act pair and the other term is used to represent both of the other state-act pairs. Such pooling equilibria will be discussed further below.

In the 4-state/4-term model, failures to approach a perfect signaling system were nevertheless observed to approach a signal success rate of about 3/4. It is curious that such systems are never observed to approach a success rate of 1/2, which would still be better than the chance expectation of 1/4 and thus might still be take to represent a language that communicates positive Shannon information.

The behavior of the 8-state/8-term model is more complicated in that there are several possible pooling equilibria corresponding to different signal success rates. The distribution of signal success rates in the 8-state/8-term model with 10^3 runs and 10^6 plays/run is given in the following table.

Signal Success Rate Interval	**Proportion of Runs**
[0.0, 0.50)	0.000
[0.50, 0.625)	0.001
[0.625, 0.75)	0.045
[0.75, 0.875)	0.548
[0.825, 1.0]	0.406

The following is an example of the contents of each of the urns for a successful run of the 3-state/3-term basic model after 10^6 plays.

Sender's Urns (for a sample successful run)

State A Urn: Ball Type	**Number of Balls**
signal 0	3
signal 1	333688
signal 2	2

State B Urn: Ball Type	**Number of Balls**
signal 0	1
signal 1	1
signal 2	332514

State C Urn: Ball Type	**Number of Balls**
signal 0	333141
signal 1	4
signal 2	20

Receiver's Urns (for a sample successful run)

Signal 0 Urn: Ball Type	Number of Balls
act A	3
act B	1
act C	333141

Signal 1 Urn: Ball Type	Number of Balls
act A	333688
act B	1
act C	4

Signal 2 Urn: Ball Type	Number of Balls
act A	2
act B	332514
act C	20

Here *term 0* corresponds to *state C* and *act C*, *term 1* corresponds to *state A* and *act A*, and *term 2* corresponds to *state B* and *act B*. Given the symmetry of the game, every permutation of terms and state-act pairs occurs.

This state is very close to an ideal signaling system since the sender and receiver will only very rarely make mistakes. The signaling success rate for this run was 0.999365 overall, the current signal success rate is better than this, and one can expect the success rate to improve since signal-act pairs will likely be successful and successful signal-act pairs will make future signal-act pairs yet more likely. Note that neither the sender nor the receiver can do better by deviating from the most likely signals.

Whenever the 3-state/3-term model does not approach a signaling system, it is observed to approach a pooling equilibrium with a signal success rate of about 2/3. The following is an example of a run that approached a pooling equilibrium.

Sender's Urns (on a sample pooling equilibrium)

State A Urn: Ball Type	Number of Balls
signal 0	186293
signal 1	3
signal 2	146829

State B Urn: Ball Type	Number of Balls
signal 0	1
signal 1	206214
signal 2	1

State C Urn: Ball Type	Number of Balls
signal 0	2

signal 1	126805
signal 2	1

Receiver's Urns (on a sample pooling equilibrium)

Signal 0 Urn: Ball Type	Number of Balls
act A	186293
act B	1
act C	2

Signal 1 Urn: Ball Type	Number of Balls
act A	3
act B	206214
act C	126805

Signal 2 Urn: Ball Type	Number of Balls
act A	146829
act B	1
act C	1

In *state A*, *term 0* is sent about 56% of the time and *term 1* is sent about 41% of the time. In *state B* and *state C*, *term 1* is almost always sent. Both *signal 0* and *signal 2* almost always lead to *act A*, and *signal 1* leads to *act B* about 62% of the time and to *act C* about 38% of the time; for a signal success rate of 0.66614 overall.

Such pooling equilibria can be characterized in general by the following diagram, where the *p* and *q* represent independent transition probabilities and a blank represents a transition probability of one.

State		**Signal**		**Act**
	p	0		
A				A
	1-*p*	1		
B			*q*	B
		2		
C			1-*q*	C

For the 3-state/3-term model, all observed pooling equilibria are permutations of this diagram. The equilibrium above, for example, might be represented by the following diagram.

State		**Signal**		**Act**
	0.56	0		

A				A
	0.41	2		
B			0.62	B
		1		
C			0.38	C

Here *p* is 0.56 and *q* is 0.62 and the expected signal success rate is about 2/3. Note that neither can do better under independent perturbations of these parameters.

Similar pooling equilibria show up in the 4-state/4-term and the 8-state/8-term models. In the latter case, there are several varieties of pooling equilibria. Such equilibria are responsible for the observed failures in convergence to signaling systems. The robustness of these pooling equilibria can be seen in the run failure rate as a function of the number of plays/run (where the signal success rate for a run failure is less than 0.8). Trials were for 10^3 runs except in the cases of 10^7 and 10^8 plays/run where the trials were 100 runs.

plays/run	run failure rate (< 0.8 signal success rate)
10^2	1.000 [10^3 runs]
10^3	0.951
10^4	0.321
10^5	0.232
10^6	0.219
10^7	0.17 [100 runs]
10^8	0.19 [100 runs]

Summary of results: While the 2-state/2-term Lewis signaling model with basic urn learning always approaches a signaling system, this is not true more generally. When the number of states is greater than two, stable pooling equilibria can evolve. Further, such pooling equilibria are more common, both in type and frequency of occurrence, the more states there are. In the 3-state/3-term model signaling systems fail to evolve approximately 9% of the time, the 4-state/4-term model fails approximately 21% of the time, and the 8-state/8-term model fails approximately 59% of the time. But, while perfect Lewis signaling systems sometimes failed to evolve, every model was always observed in these trials to evolve a language that did better than chance.

III. Lewis Signaling with Too Few and Too Many Terms

Here we consider Lewis signaling games where there are more terms than state-act pairs and where there are more state-act pairs than terms.

If there are more terms than states, this helps get convergence to a signaling system in both 3-state and 4-state models. Each extra term has a chance of evolving to connect an act with its corresponding state. The cost of this increased chance of evolving to a signaling system is in allowing for redundancy in the evolved language. With 10^6 plays/run and 200 runs, the 3-state/3-term model fails to approach a signaling system about 11% compared to a 1.5% run failure rate failure rate when there is one extra term and no observed failures when there are two extra terms. For the same parameters, the 4-state/4-term model fails to approach a signaling system about 18.5% of the time compared to a 3% run failure rate when there is one extra term and 1.5% failure rate when there are two extra terms. These results are represented in the following table.

Model	**Run Failure Rate (in approaching signaling system)**
3-state/3-term	0.120
3-state/4-term	0.015
3-state/5-term	0.000
4-state/4-term	0.185
4-state/5-term	0.030
4-state/6-term	0.015

The extra terms here evolve to cover whatever semantic gaps there may otherwise be in the coding for state-act pairs.

If there are more states than terms, then there is an information bottleneck and it is impossible for each state-act pair to be represented by a single term. The maximum signal success rate is limited by the number of terms. In the 3-state/2-term model, for example, the best possible signal success rate is 2/3. With 10^6 plays/run and 200 runs, the signal success rate was always found to approach this best possible rate.

The best possible signal success rate for the 4-state/3-term model is 3/4. With 10^6 plays/run and 200 runs, the signal success rate was always found to approach this best possible rate all but once. In the one exceptional case, it was found to approach a signal success rate of 1/2. The best possible signal success rate for the 4-state/2-term model is 1/2. This was found to evolve in every run. These results are presented in the table below.

Model	**Run Failure Rate (in approaching best possible system)**
3-state/2-terms	0.000
4-state/3-terms	0.005
4-state/2-terms	0.000

Summary of Results: Having more terms than state-act pairs helps in evolution to a signaling system, but such signaling systems invariably exhibit term redundancy. Having more state-act pairs than terms allows each term to be more efficient, but can clearly never yield a signaling system that covers each possible state.

IV. Inhomogeneous State Distributions

So far we have assumed that the states selected by nature are randomly selected with uniform probabilities. In the following 3-state/3-term models the state distribution is not uniform. Each has 10^6 plays/run and 100 runs. The state distribution [*a*, *b*, *c*] represents probability *a* of *state A*, *b* of *state B*, and *c* of *state C*, where *a*+*b*+*c*=1.

We will first consider the unbiased state distribution case for purposes of comparison.

State Distribution [1/3, 1/3, 1/3]

Signal success rate	**Proportion of runs**
1	0.92
2/3	0.08
1/3	0.00

The following are the results from trials involving inhomogeneous state distributions from most strongly biased to least strongly biased.

State Distribution [0.1, 0.1, 0.8]

Signal success rate	**Proportion of runs**
1.0	0.00
0.9	0.63
0.8	0.37
0.2	0.00

State Distribution [0.2, 0.2, 0.6]

Signal success rate	**Proportion of runs**
1.0	0.12
0.8	0.80
0.6	0.08
0.4	0.00
0.2	0.00

State Distribution [0.25, 0.25, 0.5]

Signal success rate	**Proportion of runs**
1.00	0.42
0.75	0.58
0.50	0.00
0.25	0.00

State Distribution [0.3, 0.3, 0.4]

Signal success rate	**Proportion of runs**
1.0	0.69

0.7	0.31
0.6	0.00
0.4	0.00
0.3	0.00

These simulations show that successful convergence to a signaling system is very sensitive to inhomogeneities in the distribution of states–the greater the inhomegeneity, the more likely pooling equilibria and the less likely signaling systems. In the most biased case above [0.1, 0.1, 0.8], no signaling systems at all were observed over 100 runs.

Again, it is curious that systems never seem to approach any candidate equilibria with a success rate less than 1/2. I conjecture that this is true for all Lewis signaling games with urn learning where there are at least as many terms as state-act pairs. If so, it means that more or less efficient languages always evolve in Lewis signaling games with urn learning.

Summary of results: Inhomogenious state distributions apparently increase the likelihood of pooling equilibria for Lewis signaling games with urn learning and hence discourage evolution to signaling systems.

V. Urn Learning with Positive and Negative Reinforcement

A natural modification of the basic urn learning strategy is to allow for both positive and negative reinforcement. In urn models one might think of this as adding balls of the drawn type to the urns used when the receiver's act is successful and removing balls of the drawn type from the urns used when the act is unsuccessful. Since there can never be fewer than one ball of each type in an urn, this is negative reinforcement with truncation. Punishments that would lead to fewer than one ball of each type in an urn are simply not applied.

The following 4-state/4-term models had 10^6 plays/run with 10^3 runs. Most runs had signal success rates better than 0.999. The table below displays the simulations results. The notation (+*n*, -*m*) indicates that *n* balls of the type drawn are added on success and that *m* balls of the type drawn are removed on failure. The basic urn model with only positive reinforcement is include below for comparison.

Model	**Run success rate (> 0.8 play signal success rate)**
4-state/4-term (+1, -0)	0.781
4-state/4-term (+4, -1)	0.891
4-state/4-term (+3, -1)	0.900
4-state/4-term (+2, -1)	0.960
4-state/4-term (+1, -1)	0.992
4-state/4-term (+1, -2)	0.760

Note that the 4-state/4-term models here become increasingly reliable as the ratio of the punishment for failure to reward for success increases to the point where this ratio is one.

Punishment facilitates convergence to efficient signaling by providing a mechanism for forgetting inefficient conventions that might otherwise lead to a pooling equilibrium. In the last 4-state/4-term (+1, -2) model above, the punishment for failure is so sever that it interferes with the positive reinforcement of successful strategies.

Negative reinforcement also helps in the 8-term/8-state model. Below are the results of simulations of the 8-state/8-term model with both positive and negative reinforcement. Again, there were 10^6 plays/run and 10^3 runs. The 8-term/8-state model with only positive reinforcement is included for comparison.

Model: 8-state/8-term (+1, -0)

Signal Success Rate Interval	**Proportion of Runs**
[0.0, 0.50)	0.000
[0.50, 0.625)	0.001
[0.625, 0.75)	0.045
[0.75, 0.875)	0.548
[0.825, 1.0]	0.406

Model: 8-state/8-term (+3, -1)

Signal Success Rate Interval	**Proportion of Runs**
[0.0, 0.50)	0.000
[0.50, 0.625)	0.000
[0.625, 0.75)	0.004
[0.75, 0.875)	0.225
[0.825, 1.0]	0.771

Model: 8-state/8-term (+2, -1)

Signal Success Rate Interval	**Proportion of Runs**
[0.0, 0.50)	0.000
[0.50, 0.625)	0.000
[0.625, 0.75)	0.002
[0.75, 0.875)	0.110
[0.825, 1.0]	0.888

Model: 8-state/8-term (+1, -1)

Signal Success Rate Interval	**Proportion of Runs**
[0.0, 0.50)	0.195
[0.50, 0.625)	0.116
[0.625, 0.75)	0.218

[0.75, 0.875)	0.287
[0.825, 1.0]	0.185

Here the optimal punishment to reward ratio is closer to ½. The (+3, -1) model is much better than no punishment at all, but the (+2, -1) performs yet better. The (+1, -1) model, which was the best model tested in the 4-state/4-term case, involves too much punishment to allow for positive reinforcement given the likelihood of mistakes in the evolution of the 8-state/8-term model.

Summary of results: Allowing for negative reinforcement significantly helps in converging to efficient signaling. Indeed, the 4-state/4-term (+1, -1) model is observed to almost always approach a signal system. More generally, these results show the sensitivity of Lewis signaling games to the specific details of the learning strategy.

VI. The Basic Urn Model with Random Memory

Random-memory urn models work like the basic urn models discussed above except that after each signal, act, and the possible urn reinforcement resulting from a successful act, each weight of each type of ball in each of the sender and the receiver urns is multiplied by an independent, unbiased random factor in an interval centered at 1.00. This has the effect of randomizing the records of past reinforcements. This randomization of memory effectively eliminates the pooling equilibria found in the basic urn models without mutation.

The 3-state/3-term, 4-state/4-term, and 8-state/8-term random-memory models were always found to approach a signal success rate of 1.0 on 100 runs with 10^6 plays/run and unbiased random factors in the interval [0.80, 1.20]. While the 2-state/2-term game converges without mutations, it also approaches a signal success rate of 1.0 in the random memory model with unbiased random factors in the interval [0.80, 1.20]. The signal success rate for the 2-state/2-term game at 10^6 plays/run was typically better than 0.999.

For the 3-state/3-term and the 4-state/4-term games, the distribution of run success rates was bimodal with both the lower and higher modes approaching a signal success rate of 1.0. The mean of the lower mode begins at the signal success rate associated with the pooling equilibrium, then increases with more plays. In 100 runs of the 3-state/3-term game with random factors in the interval [0.80, 1.20] and 10^6 plays/run, the success rate was typically about 0.999 with a few runs around 0.960, representing the lower mode of the distribution. In 100 runs of the 4-state/4-term game with the same parameters, most runs exhibited signal success rates of better than 0.999, but there were a few with signal success rates of about 0.971.

The bimodal character of the distribution of signal success rate and both modes mode converge to a signal success rate of 1.0 can be seen in the 4-state/4-term game with 100 runs and a random factor in the interval [0.80, 1.20].

Number of plays/run ($x10^5$)	Run failure rate (signal success rate < 0.8)
1.0	0.33
1.2	0.33
1.4	0.19
1.6	0.01
1.8	0.01
2.0	0.00

The higher the mutation rate, faster and more reliable the approach to a signal success rate of 1.0. The effect of changing the mutation rate in the 4-state/4-term game can be seen in the following table (with $2.0x10^5$ plays/run and 100 runs).

Random factor	Run failure rate (signal success rate < 0.8)
[0.80, 1.20]	0.00
[0.85, 1.15]	0.23
[0.90, 1.10]	0.24
[0.95, 1.05]	0.15
[1.00, 1.00]	0.22

The effect of increasing the number of plays per run at each of these mutation rates can be seen in the following table ($3.0x10^5$ plays/run and 100 runs).

Random factor	Run failure rate (signal success rate < 0.8)
[0.80, 1.20]	0.00
[0.85, 1.15]	0.00
[0.90, 1.10]	0.16
[0.95, 1.05]	0.17
[1.00, 1.00]	0.18

With enough runs, one always gets near convergence to a signal success rate of 1.0, even with very low mutation rates. With random factors in the interval [0.95, 1.05] and 10^7 plays/run, no run failures were observed in the 4-state/4-signal game over 100 runs. Indeed, most runs had a signal success rate better than 0.9995, the less successful mode had a signal success rate of about 0.953, and no signal success rates lower than 0.95 were observed.

It is not the case that random mutations are sufficient to generate convergence to a signaling system. Simply introducing noise to each of the sender's signals was not observed to aid in convergence to a signaling system. In order to be effective, *the random mutations must remain on the same order as the total weight of the urns.*

Summary of results: While higher mutation rates facilitate the process, every random-memory urn model was observed to approach the signaling equilibrium. Random memory strongly facilitates the evolution of signaling system on the urn learning strategy.

VII. Moran Learning

The condition in italics above can be met by Moran learning with a constant mutation rate. On a Moran learning strategy, the total weight of each urn (the number of balls in the urn) is kept constant. While this adopting such a strategy helps to discourage the occurrence of pooling equilibria, it is not, by itself fully effective. Introducing a constant mutation rate increases the chances of evolving to a signaling system. Even so, one might still converge to a new sort of pooling equilibrium if one allows for the extinction of types in the urns.

We will start with a successful model. The following 3-state/3-term Moran urn model with mutation was run 10^3 times with 10^7 plays/run and was always observed to evolve toward a near signaling system (*near signaling* because of mutations). On this model no type was ever allowed to go to extinction (one ball of each type was always left in each urn).

1. Each urn starts with a pool of 3000 balls--1000 of each term (or act) type.

2. On a successful act, add a ball of the successful type to the appropriate urn then remove one random ball, with each *type* of ball having equal probability, from each urn.

3 On each play, add one random ball and remove one random ball, with each *type* having equal probability, from the urns used in the play.

The signal success rate is typically about 0.996 on each run.

With the same learning model but without the mutation in step 3, there were 20 exceptions (signal success rate less than 0.8) in 700 runs for a success rate of about 0.971. a smaller pool size seems to help in convergence to near signaling in these models.

The following 3-state/3-term Moran urn model with mutation was always observed to evolve toward a near signaling system. On this model no type was ever allowed to go to extinction.

1. Each urn starts with a pool of 3000 balls--1000 of each term (or act) type.

2. On a successful act, add a ball of the successful type to the appropriate urn and remove one random ball, with each *ball* having equal probability, from each urn.

3. On each play there is a 0.1 chance that one random ball is added and one random ball is removed, with each *ball* having an equal probability, from the urns used in the play.

Here the signal success rate was typically about 0.85 at the 0.10 mutation rate. There were no exceptions on 377 runs with 10^7 plays/run. The signal success rate was about 0.92 when the mutation rate was lowered to 0.05. At this mutation rate, there were no exceptions on 564 runs with 10^7 plays/run. The signal success rate was about 0.98 at a mutation rate of 0.01. There were three exceptions on 800 runs at 10^7 plays/run. These exceptions, however, do not appear to be the result of a failure of the model to converge. Rather, at the lower mutations rates the convergence to a near signaling system was significantly slower and the systems were still under the success rate cutoff of 0.8.

The Moran urn model with mutation was also run *allowing for the extinction of types*. With a mutation rate of 0.05 the signal success rate was about 0.92. There are no failures to converge to a signaling system on 564 runs with 10^7 plays/run. Since unsuccessful signals and acts are forced to extinction on this model, the convergence here is absolute. The signal success rate was about 0.98 at a mutation rate of 0.01. At this lower mutation rate, there were two systems that had not yet converged to signaling on 576 runs at 10^7 plays/run.

Without the mutation in step 3 but still allowing for extinction, there were 7 convergences to pooling equilibria on 518 runs. More generally, pooling equilibria are found about 1% of the time when one stars with 1000 balls of each type in each urn. The pooling equilibria in the 3-state/3-act models exhibit a success rate of 2/3, and, because of extinction, there is no way for the system to escape from the inefficient equilibria. because of the extinction of types, these pooling equilibria have a different structure from those occurring in earlier models. An example is given in the table below.

Sender's Urns (on a sample extinction pooling equilibrium)

State A Urn: Ball Type	**Number of Balls**
signal 0	0
signal 1	3000
signal 2	0

State B Urn: Ball Type	**Number of Balls**
signal 0	3000
signal 1	0
signal 2	0

State C Urn: Ball Type	**Number of Balls**
signal 0	0
signal 1	3000
signal 2	0

Receiver's Urns (on a sample extinction pooling equilibrium)

Signal 0 Urn: Ball Type	**Number of Balls**
act A	0

act B	3000
act C	0

Signal 1 Urn: Ball Type	**Number of Balls**
act A	3000
act B	0
act C	0

Signal 2 Urn: Ball Type	**Number of Balls**
act A	0
act B	3000
act C	0

Here there simply fails to be any signal for state-act pair C.

Summary of results: Moran leaning helps in convergence to signaling. Moran learning with mutation and without extinction of types seems to guarantee convergence. Allowing for extinction of types, however, introduces a new type of suboptimal pooling equilibrium.

VIII. The Evolution of Coding and Grammar

The models discussed so far illustrate how single-term languages might evolve in a Lewis signaling game with urn learning. More complex linguistic conventions can also evolve in similar models. In the 4-state/2-term/2-sender game there are two senders who observe the state of the world, then each send a signal of either *0* or *1*. These signals are independent, and neither sender knows what the other sent. There is one receiver who cannot observe the state of the world but can observe each signal and knows which sender sent each. There are four acts, one corresponding to each of the four possible states. On the basic urn learning strategy, both of the senders and the receiver add a ball of the successful type to the appropriate urn on a successful act; and simply replace the drawn balls on failure.

Since there are fewer terms than states, a signaling system can only evolve here if the agents evolve a grammar in which to code the four possible states using sequences of the two terms. This is precisely what happens.

As with the 4-state/4-term model, the signal success rate of the 4-state/2-term/2-sender model usually approaches a signal success rate of 1.0, and when it fails to do so, it approaches a rate of 0.75. Below are the simulation results of the 4-state/2-term/2-sender model with a comparison to the results of the 4-state/4-term model.

Number of plays/run	4-state/2-term/2-sender model failure rate (< 0.8 signal rate)	4-state/4-term model failure rate (< 0.8 signal rate)

10^6	0.269 [2000 runs]	0.219 [1000 runs]
10^7	0.25 [100 runs]	0.17 [100 runs]
10^8	0.27 [300 runs]	0.19 [100 runs]

On a typical run, the 4-state/2-term/2-sender model dynamically partitions the state space and evolves a code where each sender's partial information together selects a state in the partition. The code that evolves on a successful run is a permutation of *00* means state *D*, *01* means state *C*, *10* means state *B*, and *11* means state *A*. With 10^8 plays/run, the signal success rate of the 4-state/2-term/2-sender model are typically better than 0.9999.

While the 4-state/2-term/2-sender and the 4-state/4-term models exhibit very similar behavior, it seems that pooling equilibria occur slightly more often in the 4-state/2-term/2-sender model.

More complex coding systems can be shown to evolve in models with more states. On the 8-state/2-term/3-sender model, there are eight states and three independent senders, each restricted to the terms *0* and *1*. This model converges to a signaling system about 1/3 of the time. In this case each of the eight states are represented by a three term binary string.

The distribution of signal success rates for 10^3 runs with 10^6 plays/run is given in the following table.

Signal success rate interval	**8-state/2-term/3-sender model proportion of runs**	**8-state/8-term model proportion of runs**
[0.0, 0.50)	0.000	0.000
[0.50, 0.625)	0.001	0.001
[0.625, 0.75)	0.081	0.045
[0.75, 0.875)	0.589	0.548
[0.825, 1.0]	0.329	0.406

Again, the 8-state/2-term/3-sender model does just slightly worse than the corresponding 8-state/8-term model, with one sender.

Just as in earlier models, negative reinforcement learning strategies can improve the mean signal success rate. The following table gives the results of 8-state/2-term/3-sender (+3, -1) model on 10^3 runs with 10^6 plays/run. The corresponding 8-state/2-term (+3, -1) model, with one sender, is included for comparison.

Signal success rate interval	**8-state/2-term/3-sender (+3, -1) proportion of runs**	**8-state/8-term model (+3, -1) proportion of runs**
[0.0, 0.50)	0.000	0.000
[0.50, 0.625)	0.000	0.000
[0.625, 0.75)	0.004	0.004
[0.75, 0.875)	0.157	0.225

[0.825, 1.0]	0.839	0.771

Curiously, here the 8-state/2-term/3-sender (+3, -1) model does better than its one-sender counterpart.

Summary of results: Just as single-term languages evolve in Lewis signaling games, so do languages with multi-term grammars when there are insufficient resources for a term-by-term representation of states. Both the grammar and the partition of the state space co-evolve in these models. The evolution of what one might think of as natural kinds here is in service of the rewards in learning reinforcement that accompany successful distinctions.

IX. The ARP Learning Model

The urn models discussed so far represent simple, highly idealized learning strategies, but one might also consider more sophisticated learning strategies intended to capture observed features of human learning. Bereby-Meyer and Erev (1998) describe a family of learning models that capture several salient features of human learning. The most sophisticated of these is the adjustable reference point with truncation model (ARP) model. The ARP model is also the learning model that exhibits the best overall fit to the human data presented by Bereby-Meyer and Erev. Indeed, most of the other models discussed in their study can be thought of as special cases of the ARP model.

The ARP model supports parameters that represent such features of human leaning as an agent's initial conviction, forgetfulness, and the sensitivity of the agent's dispositions to experience. It also has parameters that govern how the reward reference point moves as a function of past payoffs. On this model, the reward for a play is itself a function of the payoff and the current reference point.

The implementation of the ARP model is described in the Appendix. In agreement with Bereby-Meyer and Erev's human data, the experience generalizing factor *epsilon* is set to 0.2, the truncation parameter *v* is set to 0.0001, the forgetting parameter *phi* is set to 0.001, the initial reference point *rho* is set to 0, the positive reward weight *wpos* is set to 0.01 and the negative reward weight *wneg* is set to 0.02. The initial dispositions of the sender and receiver s[i][j] and r[i][j] are set to 27 to reflect the value that Bereby-Meyer and Erev give for S(1) in their model.

A signaling system with a signal success rate better than 0.99 was observed to evolve on every run of this model (for 617 runs total) with the parameters above and 10^7 plays/run.

The initial weights s[i][j] and r[i][j] are important to the observed success of this model. The run failure rate increased from 0.00 to 0.05 when the initial weights for s[i][j] and r[i][j] were lowered from 27 to 3. The failed runs resulted in systems with signal success rates of approximately 2/3 at the run length 10^7. But there is a trade-off here in the short-term signaling success rate. While higher initial weights for s[i][j] and r[i][j] make it more likely that one gets a signaling system in the long-term, short-term signaling success rates are typically higher with lower initial weights for s[i][j] and r[i][j].

Summary of results: There are plausible human parameters for the ARP learning model that lead to signaling systems in every observed simulation. This explains why human agents might be expected to evolve signaling systems in the context of a Lewis signaling games. It also illustrates how the evolution of signaling systems in the Lewis signaling game ultimately depends on the learning strategy.

X. Conclusion

While the 2-state/2-term Lewis signaling game with basic urn learning always approaches a signaling system, pooling equilibria corresponding to suboptimal success rates are increasing common in both type and frequency of occurrence when there are more than two states. Biased state distributions also increase the likelihood of pooling equilibria and thus discourage evolution to signaling systems. There are however, several features that facilitate convergence to signaling.

Having more terms than state-act pairs helps in evolution to a signaling system. Negative reinforcement in a learning strategy can significantly aid in convergence to efficient signaling. Processes that generate mutations that are on the order of the current weight of reinforcement and Moran learning also strongly facilitate the evolution toward a signaling success rate of 1.0 urn learning strategies. And finally, there are plausible human parameters for the ARP learning model that also lead to signaling systems in every observed simulation.

Just as single-term languages can evolve in Lewis signaling games, so also can multi-term languages with a simple grammar. The terms, the grammar, and the corresponding partition of the state space co-evolve to given the appearance of natural kinds with a corresponding language. The apparent natural kinds, however, are simply in service of the rewards that accompany successful action in the model. Metaphysical morals drawn from the structure of a natural language that evolved in this way would only track arbitrary, but pragmatically useful distinctions.

Appendix

The implementation of the ARP model is described below. In agreement with Bereby-Meyer and Erev's human data, the experience generalizing factor *epsilon* is set to 0.2, the truncation parameter *v* is set to 0.0001, the forgetting parameter *phi* is set to 0.001, the initial reference point *rho* is set to 0, the positive reward weight *wpos* is set to 0.01 and the negative reward weight *wneg* is set to 0.02. The initial dispositions of the sender and receiver s[i][j] and r[i][j] are set to 27 to reflect the value of S(1) on Bereby-Meyer and Erev's model.

```
s[3][3];     // sender dispositions [state][signal]
r[3][3];     // receiver dispositions [signal][act]
signal;      // 0 to 2: three possible signals
act;         // 0 to 2: three possible acts
state;       // 0 to 2: three possible states
rtest;       // test random number for urn draws
run;         // number of runs so far
exceptions; // number of imperfect runs
epsilon;     // experience generalizing factor
v;           // truncation parameter
phi;         // forgetting parameter
rho;         // moving reference point
wpos;        // positive weight
wneg;        // negative weight


//check for success of act and update sender and receiver dispositions
if(act==state)
      {
      // successful signal

      //update reference point
      if(rho<=1)
            rho = ((1-wpos)*rho) + wpos;
      else
            rho = ((1-wneg)*rho) + wneg;

      s[state][signal] = ((1-phi)*s[state][signal])
                              + ((1-rho)*(1-epsilon));

      r[signal][act] = ((1-phi)*r[signal][act])
                              + ((1-rho)*(1-epsilon));
      }
else
      {
      // failed signal

      //update reference point
      if(rho<=0)
                  rho = ((1-wpos)*rho);
      else
                  rho = ((1-wneg)*rho);

      s[state][signal] = ((1-phi)*s[state][signal]) – (rho*(epsilon));

      r[signal][act] = ((1-phi)*r[signal][act]) – (rho*(epsilon));
      }

//truncate diposition weights if necessary
if(s[state][signal] < v)
      s[state][signal]=v;
```

```
if(r[signal][act] < v)
      r[signal][act]=v;
```

The initial sender and receiver dispositional weights s[i][j] and r[i][j] determine Bereby-Meyer and Erev's parameter S(1). The weights s[i][j] are subsequently used to determine the sender's stochastic signal and the weights r[i][j] are used to determine the receiver's stochastic act.

Acknowledgements: I would like to thank Brian Skyrms for introducing me to this subject and for many productive conversations. I would also like to thank Rory Smead and Kevin Zolman for several insightful suggestions. Smead is currently running population models similar to the leaning models discussed in this paper.

References

Bereby-Meyer, Yoella and Ido Erev (1998) "On Learning to become a Successful Loser: A Comparison of Alternative Abstractions of Learning Processes in the Loss Domain, *Journal of Mathematical Psychology* **42**:226-286.

Lewis, David (1969) *Convention* Cambridge, Mass: Harvard University Press.

Huttegger, Simon (forthcoming A) "Evolution and the Explanation of Meaning" *Philosophy of Science.*

Huttegger, Simon (forthcoming B) "Evolutionary Explanations of Indicatives and Imperatives" *Erkenntnis.*

Skyrms, Brian (2004) *The Stag Hunt and the Evolution of Social Structure* Cambridge: Cambridge University Press.

Skyrms, Brian (2006) "Signals" Presidential Address Philosophy of Science Association for PSA 2006