Learning to Signal in a Dynamic World

Dr J McKenzie Alexander

Department of Philosophy, Logic and Scientific Method
London School of Economics and Political Science

20 April 2011

Outline

1. Introduction: Lewis signaling games and the move to reinforcement learning

2. Inventing new signals with forgetting

3. Coping with a dynamic environment

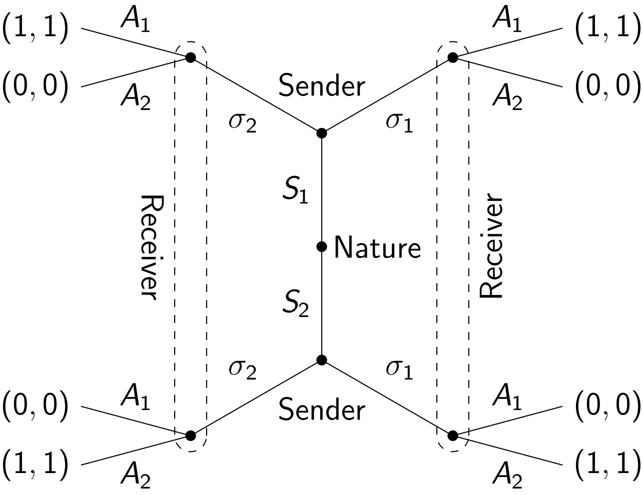
4. Inventing new signals with discounting

5. Conclusion

Outline

1. Introduction: Lewis signaling games and the move to reinforcement learning
2. Inventing new signals with forgetting
3. Coping with a dynamic environment
4. Inventing new signals with discounting
5. Conclusion

Lewis Sender-Receiver games



The emergence of meaning via reinforcement

Regarding the two-state, two-signal and two-action game (states equiprobable), Skyrms writes:

> "Spontaneous emergence of signaling [. . .] requires no strategic reasoning, just chance and reinforcement. This is, in fact, just what happens. Individuals *always* learn to signal in the long run. This is not only confirmed by extensive simulations, it is also a theorem."
>
> (Skyrms, 2010, pg. 94)

Learning to signal

Regarding more complicated problems, Skyrms says:

> “How hard is it to learn to signal? This depends on our criterion of success for the learning rule. If success means spontaneous generation of signaling in many situations, then all the kinds of learning that we have surveyed pass the test [. . .] If it means learning to signal with probability one in all Lewis signaling games, a simple payoff-based learning rule will do the trick. It is easy to learn to signal.”
>
> (Skyrms, 2010, pg. 105)

Two key assumptions, and two questions

Skyrms assumes that the structure of the Lewis signaling game is *fixed*. The number of state-action pairs is taken as given, as is the number of possible signals (except for chapter 10).

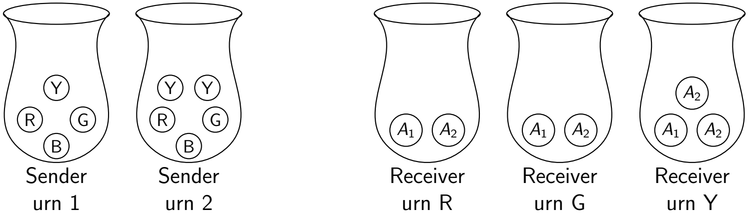
Reinforcement learning then operates inside this setting.

1. Can efficient, minimal signaling systems emerge *ex nihilo* in a fixed sender-receiver game?
2. If so, what then happens if the structure of the game is made *dynamic*, varying the number of state-action pairs, or what counts as a "correct response"?

Outline

1. Introduction: Lewis signaling games and the move to reinforcement learning
2. Inventing new signals with forgetting
3. Coping with a dynamic environment
4. Inventing new signals with discounting
5. Conclusion

Inventing new signals: The Hoppe-Pólya Urn

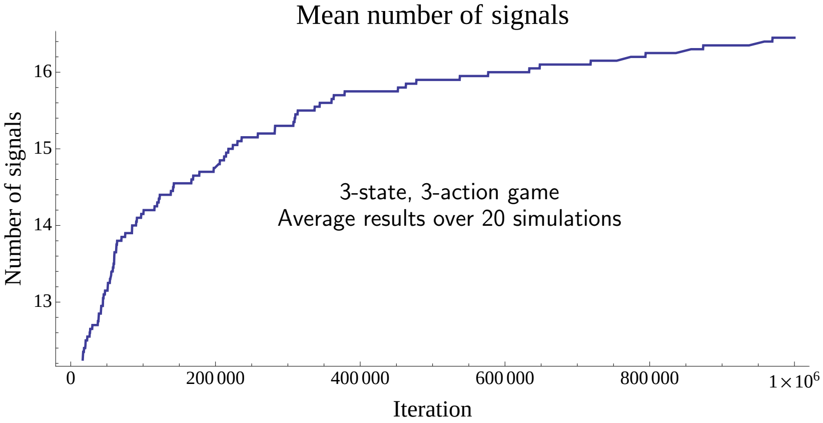


After successful invention

- State of the world: 2.
- The Sender draws the black ball from urn 2 and hence tries out a new signal Y.
- The Receiver creates a new urn for the signal. Suppose she draws, by chance, A_2. This is the correct act.
- Reinforcement occurs.

Inventing new signals

This model of invention generates signaling systems, but many unnecessary signals are also generated.



Inventing with Forgetting

Skyrms et al. (2011) augment this model of inventing signals with forgetting to see if minimal, as well as efficient, signaling systems can be generated.

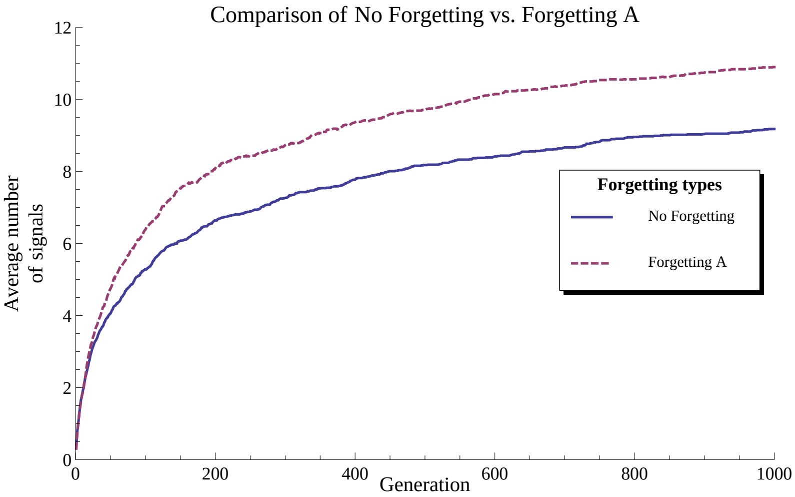
Forgetting A. Periodically select an urn at random, then select a ball from the urn and throw it away.

Forgetting B. Periodically select an urn at random, then select a *colour* found in *that* urn, and throw away one ball of that colour.

In both cases, the mutator ball is excluded.

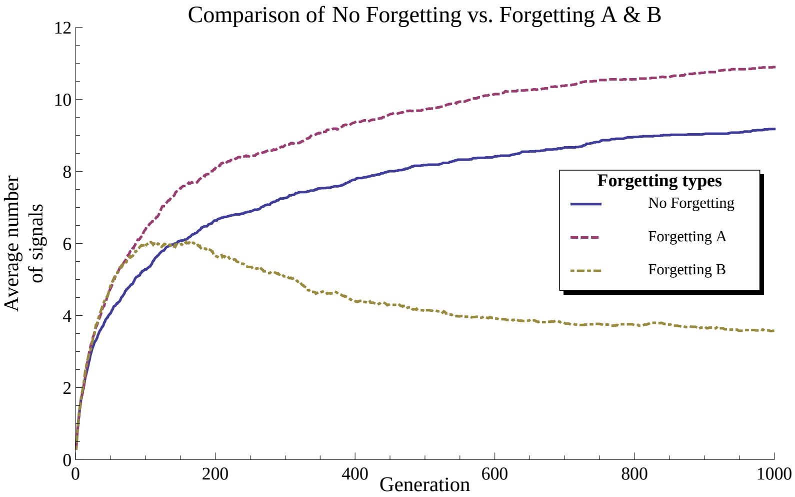
“Forgetting A” is worse than “No Forgetting”

3 states, 3 actions, all states equally likely.



Comparison of both forgetting methods

3 states, 3 actions, all states equally likely.



Average number of signals remaining

Simulation results: equiprobable states, each entry the average of 1,000 simulations run for 1,000,000 iterations each, forgetting rate of $\frac{1}{3}$.

	No Forgetting	Forgetting A	Forgetting B
3 states	16.276	19.879	3.016
4 states	17.491	21.079	4.005
5 states	18.752	22.686	4.982
6 states	20.097	24.069	5.975
7 states	21.336	25.82	6.96
8 states	22.661	27.14	7.941
9 states	23.815	28.684	8.929
10 states	24.925	30.663	9.928

Inventing new signals: a 30-state, 30-action game

Sender

Signals

	0	137	343	398	475	620	1098	1119	1457	1615	1846	2251	2501	2951	3634	3750	6711	6889	7753	7849	9124	13 937	18 816	19 930	22 893	24 774	32 283	43 151	65 425	75 275	86 715
State 1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45 697	0	0	0	0	0	0	0	0	0	0	0
State 2	1	0	0	0	0	0	0	0	46 726	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46 616	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46 450	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 5	1	0	0	0	0	46 969	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	43 088	0	0	0	0	0
State 7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45 432	0	0	0	0	0	0	0	0	0	0
State 8	1	0	0	0	0	0	0	0	0	0	47 008	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 9	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45 789	0	0	0	0	0	0	0	0	0	0	0	0
State 10	1	0	0	0	47 075	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	44 942	0	0	0	0	0	0	0	0	0
State 12	1	0	0	0	0	0	0	46 553	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 13	1	0	47 017	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 14	1	0	0	0	0	0	0	0	0	0	0	0	46 438	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	46 201	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 16	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	44 316	0	0	0	0	0	0	0
State 17	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	43 893	0	0	0	0	0	0	0	0
State 18	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39 107	0	0	0
State 19	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	18 663
State 20	1	0	0	0	0	0	0	0	0	0	0	0	0	46 770	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 21	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	43 820	0	0	0	0	0	0
State 22	1	0	0	0	0	0	0	0	0	0	0	47 092	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 23	1	46 888	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 24	1	0	0	46 446	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 25	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45 877	0	0	0	0	0	0	0	0	0	0	0	0	0
State 26	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31 657	0	0
State 27	1	0	0	0	0	0	46 011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 28	1	0	0	0	0	0	0	0	0	46 295	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
State 29	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27 523	0
State 30	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	41 622	0	0	0	0

Receiver

	Act 1	Act 2	Act 3	Act 4	Act 5	Act 6	Act 7	Act 8	Act 9	Act 10	Act 11	Act 12	Act 13	Act 14	Act 15	Act 16	Act 17	Act 18	Act 19	Act 20	Act 21	Act 22	Act 23	Act 24	Act 25	Act 26	Act 27	Act 28	Act 29	Act 30
Signal 137	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	66 240	1	1	1	1	1	1	1
Signal 343	2	1	1	1	1	1	1	1	1	1	1	1	66 498	1	1	1	1	1	2	1	1	1	1	1	1	1	2	1	1	1
Signal 398	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	65 884	1	1	1	1	1	1
Signal 475	1	1	1	2	1	1	1	1	1	66 327	1	1	1	1	1	2	1	1	1	1	1	1	1	1	2	1	1	2	1	1
Signal 620	1	1	1	1	66 376	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1
Signal 1098	1	1	1	2	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	2	1	1	1	1	1	1	65 481	1	1	1
Signal 1119	1	1	1	1	1	1	1	1	1	1	1	65 920	1	2	1	1	1	2	1	2	1	1	1	1	1	1	1	1	1	1
Signal 1457	1	66 083	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1
Signal 1615	1	1	1	17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	10	1	1	1	1	1	1	1	65 807	1	1
Signal 1846	1	1	1	1	1	1	1	66 117	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 2251	1	1	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	66 204	1	1	1	1	1	1	1	1
Signal 2501	1	1	1	1	1	1	1	1	1	1	1	1	1	65 563	108	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 2951	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	66 175	1	1	1	1	1	1	1	1	1	4
Signal 3634	1	1	1	1	1	3	1	1	1	1	1	1	1	1	65 387	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 3750	2	1	66 083	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1
Signal 6711	2	1	1	65 541	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 6889	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	65 083	1	1	1	1	1
Signal 7753	1	1	1	1	1	3	1	1	65 072	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 7849	64 877	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 9124	1	1	1	1	1	1	64 715	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 13937	1	1	1	1	1	1	1	1	1	1	63 872	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 18816	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	62 856	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 19930	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	63 149	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 22893	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	62 705	1	1	1	1	1	1	1	1	1
Signal 24774	1	1	1	1	1	62 179	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Signal 32283	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	60 009
Signal 43151	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	56 538	1	1	1	1	1	1	1	1	1	1	1	1
Signal 65425	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	46 704	1	1	1	1
Signal 75275	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	40 849	1

Summary, so far

The model of inventing signals, with Forgetting B, is *extremely effective* at arriving at efficient, minimal signaling systems.

So is it really true, then, that "it is easy to learn to signal"?

Forgetting with unequal state probabilities

Consider a 3-state, 3-action game with Forgetting B rate p.

Each iteration, there's a $\frac{p}{3}$ chance of a ball being thrown out of the Sender's urn used in state i.

If the probability of state i is *less* than $\frac{p}{3}$, reinforcement learning won't occur fast enough to overcome the rate at which balls are removed!

Forgetting with unequal state probabilities

Illustration

3-state, 3-action game with Forgetting B rate $\frac{1}{3}$.

State probabilities: $\left\langle \frac{4}{9} - \frac{\varepsilon}{2}, \frac{4}{9} - \frac{\varepsilon}{2}, \frac{1}{9} + \varepsilon \right\rangle$ with $\varepsilon = \frac{1}{10{,}000}$.

		Sender		
		Signals		
	0	14	18	223 714
State 1	1	6 744 670	0	0
State 2	1	0	6 737 099	0
State 3	1	0	0	831

		Receiver	
		Actions	
	Act 1	Act 2	Act 3
Signal 14	8 892 328	1	1
Signal 18	2	8 884 925	1
Signal 223714	1	1	1 031 869

After 20,000,000 iterations

But there's a greater problem concerning learning to signal. . .

Outline

1. Introduction: Lewis signaling games and the move to reinforcement learning

2. Inventing new signals with forgetting

3. **Coping with a dynamic environment**

4. Inventing new signals with discounting

5. Conclusion

Learning to signal in a dynamic world

All of the cases discussed in *Signals* use a fixed number of state/action pairs.

What happens if the environment is dynamic? That is,

1. What if new state-action pairs are added over time?
2. What if the correct response to a given state of the world is swapped with another state?

Adding new states

Let q be the probability that a new state-action pair is added to the model at the start of a given iteration.

Suppose we have an N-state, N-action signaling game with state probabilities of $\langle p_1, \ldots, p_N \rangle$.

If a state-action pair is added, the new state probabilities are:

$$\left\langle \frac{N}{N+1} p_1, \ldots, \frac{N}{N+1} p_N, \frac{1}{N+1} \right\rangle$$

Adding new states II

The choice of $\frac{N}{N+1}$ as the renormalization factor ensures that if we begin with equiprobable states, we continue to have equiprobable states.

$$\left\langle \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right\rangle \xrightarrow{\text{New state}} \left\langle \frac{3}{4}\cdot\frac{1}{3}, \quad \frac{3}{4}\cdot\frac{1}{3}, \quad \frac{3}{4}\cdot\frac{1}{3}, \quad \frac{1}{4} \right\rangle$$

$$= \left\langle \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4} \right\rangle$$

A simulation result

Beginning with a 1-state, 1-action model with an addition probability of 0.001:

After 5,000 iterations

	Sender									
	Signals									
	0	2	9	22	69	85	86	102	187	225
State 1	1	1227	0	0	0	0	0	0	0	1
State 2	1	0	708	0	0	0	0	0	0	1
State 3	1	0	0	466	0	0	0	0	0	0
State 4	1	0	0	0	0	169	0	0	0	1
State 5	1	0	0	0	158	0	0	0	0	1
State 6	1	0	0	0	0	0	175	0	0	0
State 7	1	0	0	0	0	0	0	117	0	1
State 8	1	0	0	0	0	0	0	0	11	0
State 9	1	0	0	0	0	0	0	0	0	2

	Receiver								
	Actions								
	Act 1	Act 2	Act 3	Act 4	Act 5	Act 6	Act 7	Act 8	Act 9
Signal 2	1794	1	1	1	1	1	1	1	1
Signal 9	1	972	1	1	1	1	1	1	1
Signal 22	1	1	684	1	1	1	1	1	1
Signal 69	1	1	1	1	264	1	1	1	1
Signal 85	1	1	1	248	1	2	1	1	1
Signal 86	1	1	1	1	1	245	1	1	1
Signal 102	1	1	1	1	1	1	186	1	1
Signal 187	1	1	1	1	1	1	1	34	2
Signal 225	1	1	1	1	1	1	1	1	2

Measuring how effectively signals are invented

Let $\vec{s} = \langle s_i \rangle_{i=1}^{\infty}$ denote the sequence of outcomes, where

$$s_i = \begin{cases} 1 & \text{if the signaling attempt was successful,} \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$\frac{\sum_{i=1}^{k} s_i}{k}$$

measures the cumulative frequency of signaling success up to iteration k, and

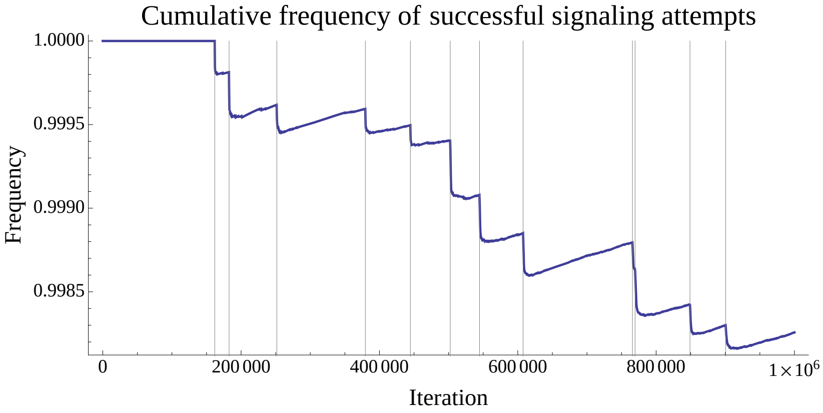
$$\frac{\sum_{i=k-100}^{k} s_i}{k}$$

measures the "moving frequency" of signaling success over the past 100 iterations at k.

Simulation results

Initial model: 1-state, 1-action signaling game, forgetting rate 0.333

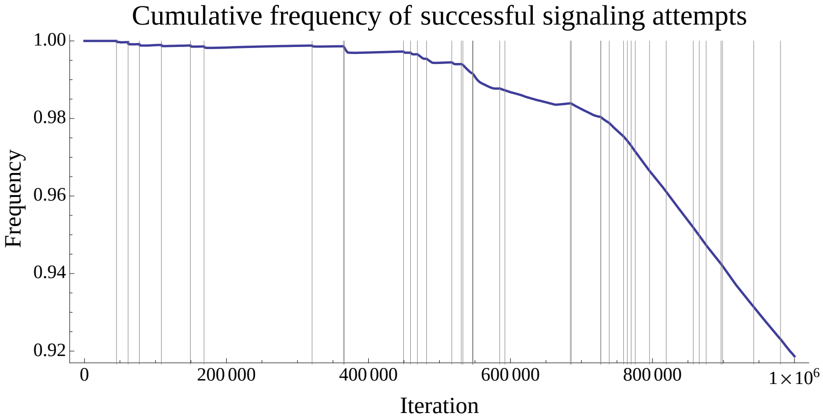
New state probability 0.00001.



A simulation result

Initial model: 1-state, 1-action signaling game, forgetting rate 0.333

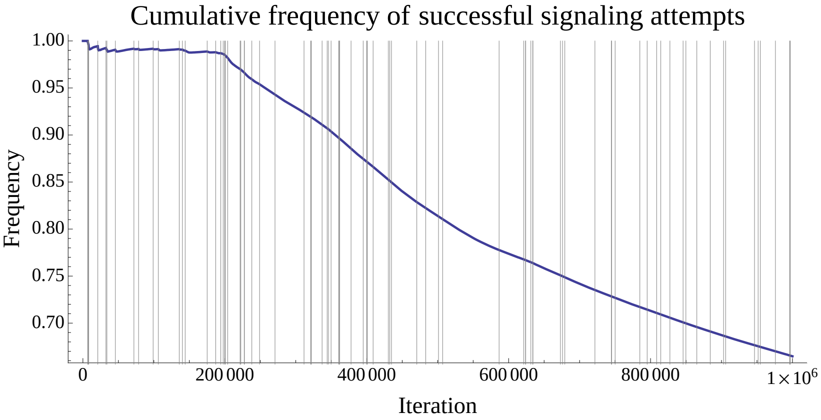
New state probability 0.0000333.



A simulation result

Initial model: 1-state, 1-action signaling game, forgetting rate 0.333

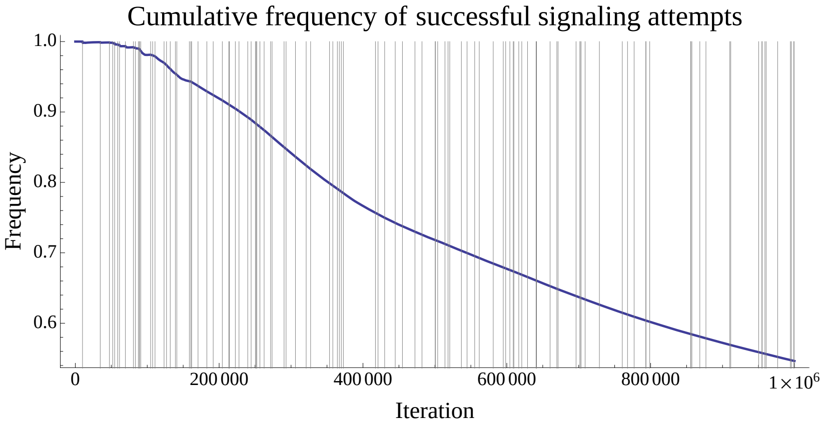
New state probability 0.0000666.



A simulation result

Initial model: 1-state, 1-action signaling game, forgetting rate 0.333

New state probability 0.0001.



Swapping states

Let p be the probability that, at the start of each iteration, the correct response to a given state of the world *permanently* switches.

Think of this as a change in the fitness landscape due to elements beyond the control of the Sender and Receiver.

Example

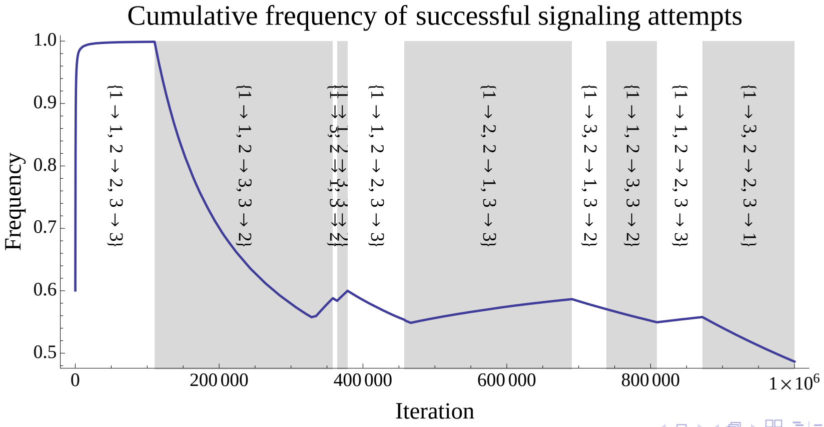
Before:	State 1→ Action 1	After:	State 1→ Action 3
	State 2→ Action 2		State 2→ Action 2
	State 3→ Action 3		State 3→ Action 1

How well does inventing with Forgetting B cope?

A simulation result

3-state, 3-action signaling game, forgetting rate 0.333, swap rate 0.00001

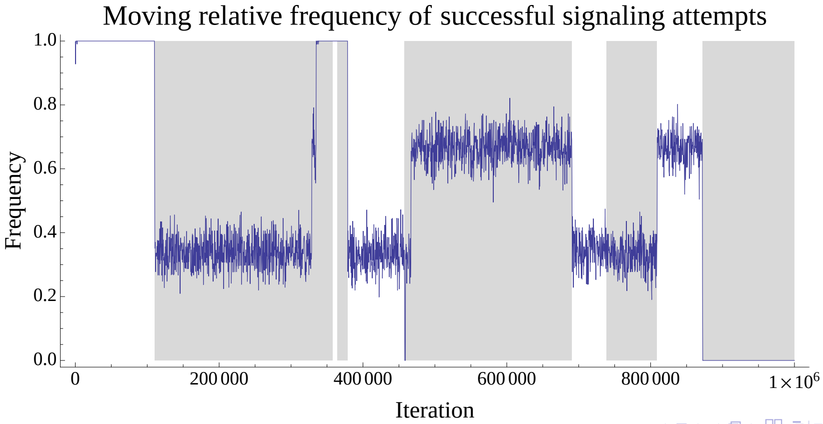
In a dynamic environment, the model of Skyrms et al. (2011) takes too long to “unlearn” a signaling system.



A simulation result

3-state, 3-action signaling game, forgetting rate 0.333, swap rate 0.00001

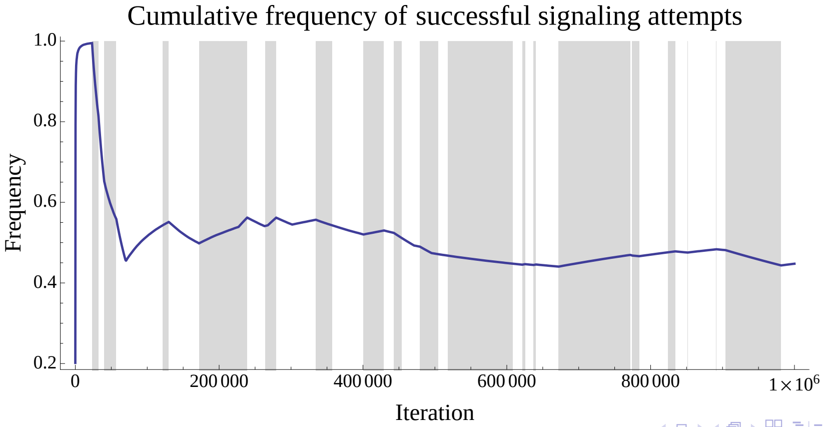
Once the initial signaling system is learnt, the overall performance is effectively determined by chance:



A simulation result

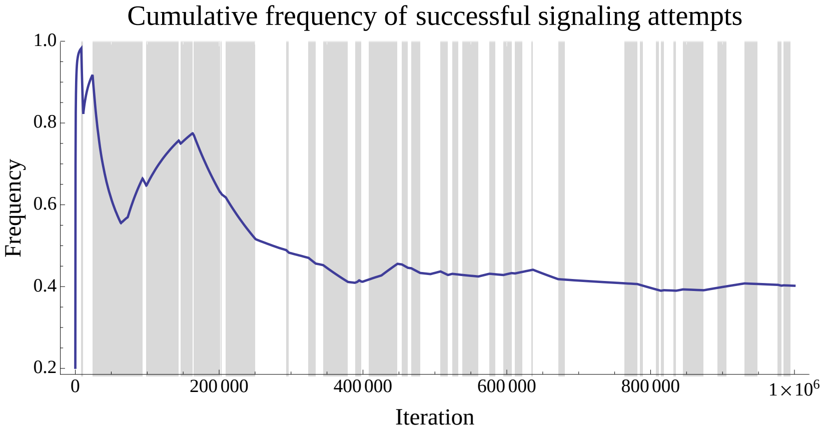
3-state, 3-action signaling game, forgetting rate 0.333, swap rate 0.0000333

Increasing the dynamic element of the signaling problem drives the model's performance to effectively that of chance:



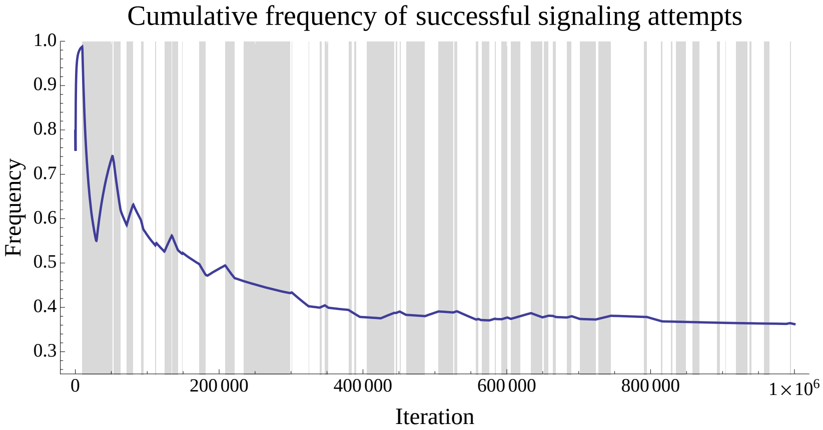
A simulation result

3-state, 3-action signaling game, forgetting rate 0.333, swap rate 0.0000666



A simulation result

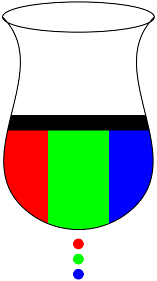
3-state, 3-action signaling game, forgetting rate 0.333, swap rate 0.0001



Outline

1. Introduction: Lewis signaling games and the move to reinforcement learning
2. Inventing new signals with forgetting
3. Coping with a dynamic environment
4. Inventing new signals with discounting
5. Conclusion

The leaky Hoppe-Pólya urn



Consider a Hoppe-Pólya urn, but filled with liquids rather than discrete balls.

A black liquid, which floats on the others, corresponds to the "mutator" ball.

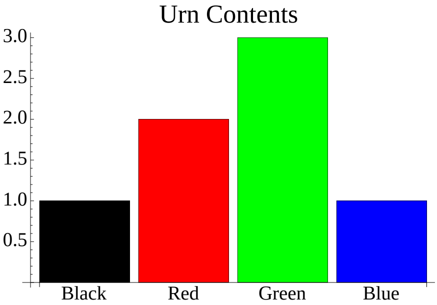
Reinforcement occurs by pouring a cup of coloured liquid into the urn.

But the urn is leaky, so over time the non-black coloured liquids drain away.

The leaky Hoppe-Pólya urn

This is a model of reinforcement with *discounting*. (Except for the black liquid.)

The discount rate is how quickly the urn drains.



Discount rate of 0.75

The leaky Hoppe-Pólya urn

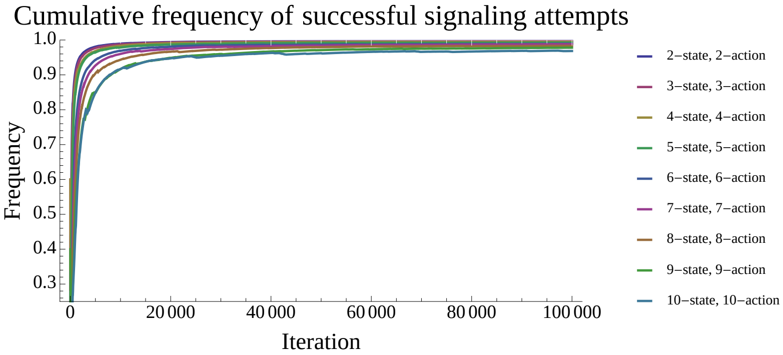
Because discounting only assigns probability 0 to a signal in the limit, a *cutoff threshold* τ is introduced.

If the probability of a signal dips below τ, that signal is eliminated.

Discounting also applies to the *Receiver's* urns — except that the cutoff threshold doesn't apply. (Why? Signals are purely conventional, but actions are not.)

Inventing new signals with discounting

Inventing with discounting does nearly as well as inventing with forgetting (here, the discount rate is 0.95):



Inventing new signals with discounting

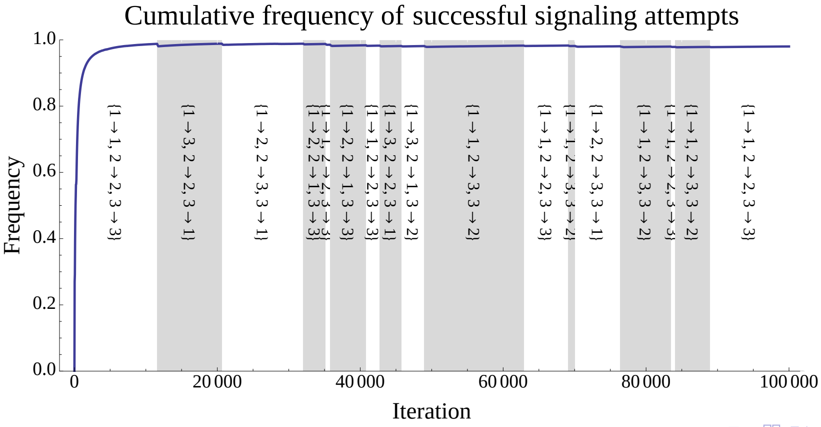
Efficiency considerations, discount rate: 0.95

Game type	Mean number of signals
2-states, 2-actions	3.47958
3-states, 3-actions	4.29091
4-states, 4-actions	5.21725
5-states, 5-actions	6.21697
6-states, 6-actions	7.211
7-states, 7-actions	8.19678
8-states, 8-actions	9.04261
9-states, 9-actions	9.13345
10-states, 10-actions	8.03297

Inventing with discounting: swapping states

3-state, 3-action game, discount rate 0.95, swap probability 0.0001.

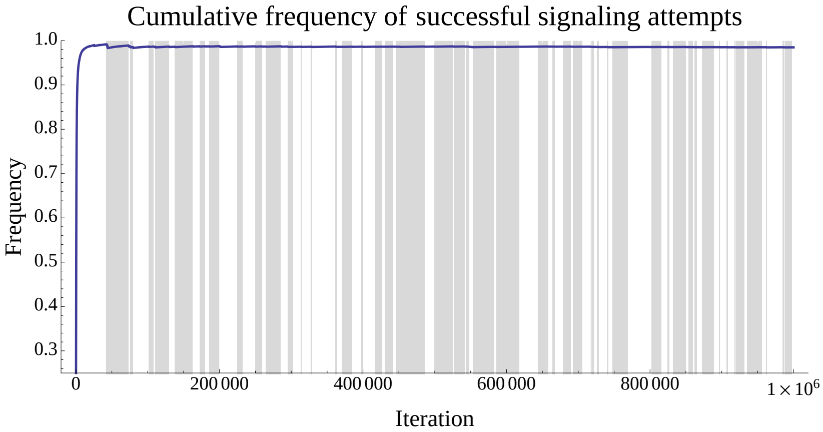
Unlike the standard urn model, discounting responds rapidly to swapped states:



Inventing with discounting: swapping states

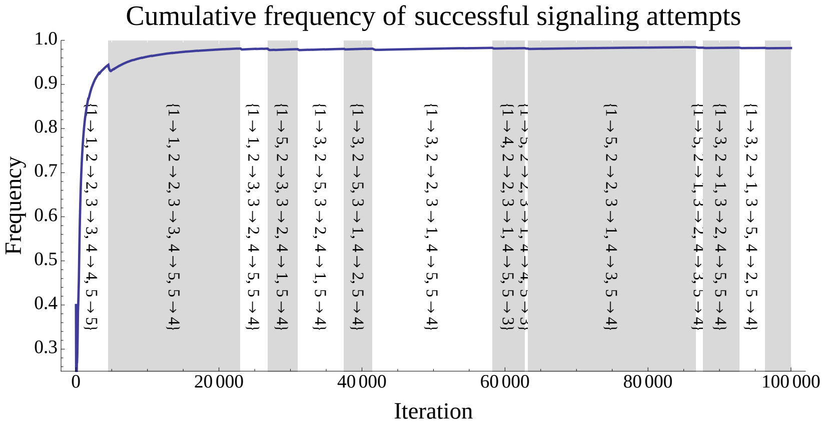
3-state, 3-action game, discount rate 0.95, swap probability 0.0001.

One million iterations of the discount model:



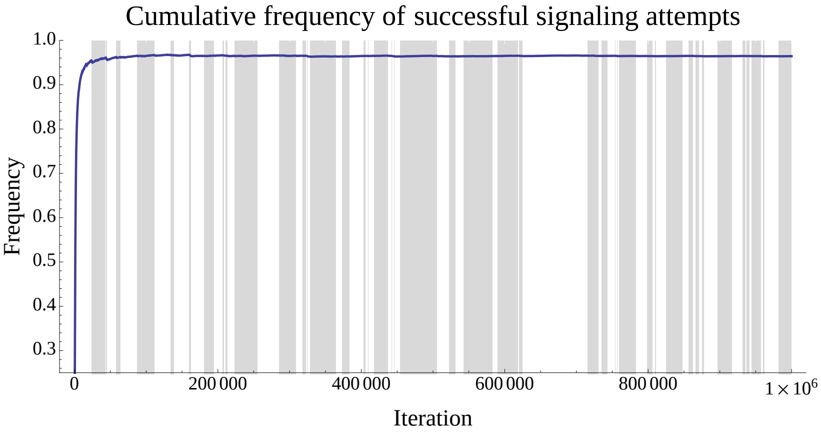
Inventing with discounting: swapping states

5-state, 5-action game, discount rate 0.95, swap probability 0.0001.



Inventing with discounting: swapping states

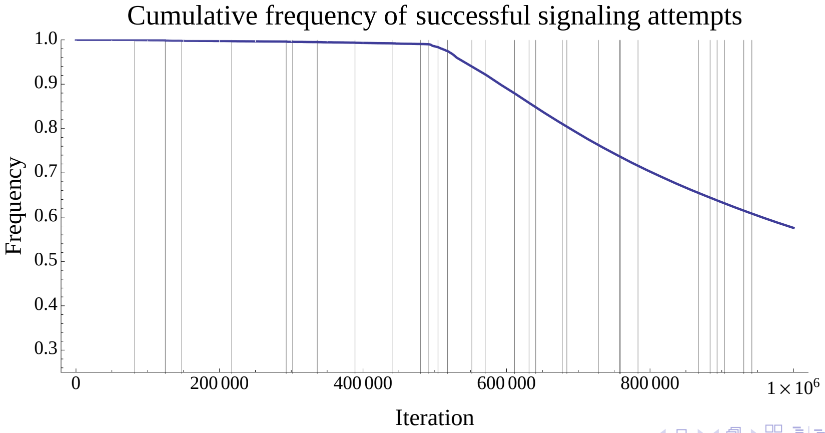
10-state, 10-action game, discount rate 0.95, swap probability 0.0001.



Inventing with discounting: adding new states

Initial game: 1-state, 1-action, new state probability 0.000333, discount rate 0.95

Unfortunately, the discounting model is little better at handling the addition of new states than the original model.



Efficient drift in signaling systems

In real languages, drift occurs, even though effective communication occurs at each point in time.

Whan that Aprille with his shoures soote
The droghte of Marche hath perced to the roote,
And bathed every veyne in swich licour,
Of which vertu engendred is the flour;

(Chaucer)

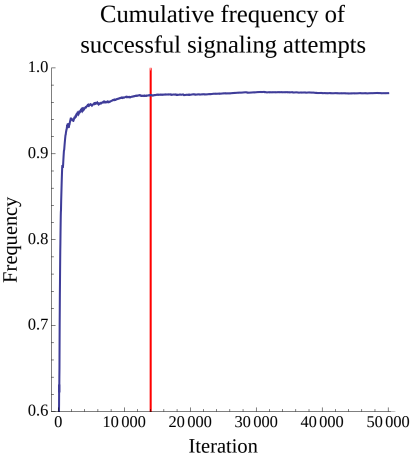
Is it thy will, thy image should keep open
My heavy eyelids to the weary night?
Dost thou desire my slumbers should be broken,
While shadows like to thee do mock my sight?

(Shakespeare)

Fa' shizzle dizzle
It's Big Snoopy D-O double Gizzle
Wit my nephew Swizzle, you know it's off the hizzle

(Snoop Dogg)

Signaling drift while maintaining efficiency



	Sender			
	Signals			
	0	97	223	229
State 1	0.1	0	8.2927	0
State 2	0.1	3.22687	0	0
State 3	0.1	0	0	6.40263

	Receiver		
	Actions		
	Act 1	Act 2	Act 3
Signal 97	0	3.22687	0
Signal 223	8.2927	0	0
Signal 229	0	0	6.40263

Outline

1. Introduction: Lewis signaling games and the move to reinforcement learning

2. Inventing new signals with forgetting

3. Coping with a dynamic environment

4. Inventing new signals with discounting

5. Conclusion

Conclusion

- The model of inventing signals with forgetting in *Signals* (and Skyrms et al., 2011) effectively generates efficient, minimal signaling systems in many cases.
- Yet this model performs poorly in dynamic environments.
- Inventing with discounting has improved performance in one kind of dynamic environment (swapping).
- Inventing with discounting also allows efficient drift between signaling systems over time.
- How to cope with the problem of adding new state/action pairs remains an open question.

References and miscellaneous readings I

Brian Skyrms. *Signals: Evolution, Learning, & Information*. Oxford University Press, 2010.

Brian Skyrms, Sandy Zabell, and J. McKenzie Alexander. Inventing signals with forgetting. *Dynamic Games and Applications*, 2011.

Outline

6 Simulation results