Investigating Language Change: A Multi-Agent Neural-Network Based Simulation

Scott C. Stoness (scs@sfu.ca)
School of Computing Science
Simon Fraser University

Christopher Dircks (chrisd@sfu.ca)
School of Computing Science
Simon Fraser University

Abstract

Multiple agents, equipped with a feature-based phonetic model and a connectionist cognitive model, interact via the *naming game* , with lexicon formation and change as emergent properties of this complex adaptive system. We present a new description of the *naming game*, situating it as a general, implementation-independent paradigm. Our addition of richer phonetic and cognitive models provides the agents with a greater degree of cognitive validity than does earlier work, while enhancing the flexibility of the system and reproducing empirical results. Feature-based phonetics, piecewise reinforcement learning, and a connectionist architecture with local representation allows language discrimination based on schemata instead of entire utterances.

Introduction

All things change. Despite societal and personal predispositions towards stability, constant modification is an incontrovertible fact, irrespective of the observed impact on our quotidian existence. We may not be cognizant, as individuals, of the process of change as it occurs, because the brief time span of a human life does not permit the recognition of developments preceding great change; events do not always match the meteoric pace set by the limits of our frail biology, but rather transpire with a certain glacial implacability. One phenomenon with which we are all intimately acquainted that undergoes just such an incremental process of adjustment is natural language; the idiolect of any single individual remains relatively fixed subsequent to the initial acquisition of the mother tongue, yet the language as a whole clearly experiences periodic alteration.

It is no coincidence that the description above of a gradual, gradational modification also applies to evolution by natural selection, since language change shares many of the basic attributes of its biological parallel. The deliberate nature of this type of process can obfuscate the situation from an individual perspective. This tendency was exacerbated in the late 19th century by the conflation of language change with an incomplete comprehension of the biological principle – obsession with progress, and the usurption for this goal of the phrase 'survival of the fittest' – leading to the simultaneous adoption of both historical reconstruction, which would map out the myriad developments in the evolution of language *and* prescriptive grammar, where the stated goal is to maintain the purity of the current linguistic norms.

20th century linguists have relinquished their grip on the reins and recognized the inevitability of language change, but have been unable to converge upon a single theory, or more accurately, have been unable to produce a universally compelling explanatory mechanism for language change. It puts modern linguists in much the same position as biologists before the advent of Darwin's theory of natural selection, or perhaps before the discovery of DNA by Watson & Crick; there is a universal acceptance of a general process, but the details are not known.

Unfortunately, the data available for historical research is limited since the vast majority of the world's languages spoken to date had no written form, and much of what was committed to paper(or other appropriate media) has been lost. Controlled linguistic experimentation is extraordinarily difficult, the more so when the phenomenon we are examining would require investigations which would not begin to bear fruit for several generations. The linguistic analogue to *Drosophilia* is not obvious; there simply are no biological entities which exhibit sufficient similarities to human linguistic communication to make experimentation worthwhile *and* are also short-lived enough for such experiments to be feasible. Fortunately, modern computers are sufficiently powerful to enable us to produce simulations of language change which are, to a certain degree, cognitively and linguistically accurate, yet simplistic enough to allow controlled investigation of this phenomenon.

Much recent work has been done in this vein, in particular by the members of the Sony CSL Paris, examining language as a complex adaptive system which produces language change as emergent behaviour. [9] postulates that four factors are necessary for linguistic variation and evolution to occur: self-organization, stochasticity in transmission and production, tolerance of minor linguistic variation, and a certain rate of population change. The authors first produced completely deterministic agents, showing that linguistic variation was not tolerated. As a result, the final model (reported in [8], [9], and prototypically in [7]) of agent-internal linguistic processing is something of a hybrid, with probabilistic measures grafted onto what is essentially a deterministic backbone, while linguistic utterances in the system are represented by a random sequence of characters. We present a new system for investigating lexicon change which follows a framework for agent interaction similar to the Steels and Kaplan model; however, we have made our system more valid from a cognitive perspective by using artificial neural networks for the agent-internal cognitive model and a phonetic model based on Chomsky-Halle binary features for utterances.

Agent Interactions: The Naming Game

It is foolhardy to undertake research involving multiple software agents without a principled framework within which the agents can interact; that is, the precise details of all possible interactions between agents and their environment (and / or each other) must be specified in an unambiguous fashion. To this end, [7] introduces the *naming game*, an austere paradigm for interaction tailored especially for the development, transmission, and evolution of a lexicon in either a static or dynamic population of agents.

The *naming game* is appropriate for a population of agents and a number objects[1]; an interaction proceeds as follows:

1) Two agents are selected from the population; one is designated as the Speaker, the other as the Hearer.
2) The Speaker chooses an object, possibly at random.
3) The Speaker, through whatever process encoded in the agent model, names the object; that is, accesses the appropriate form-meaning pair, and produces the form.
4) The Speaker (virtually) points at the object.
5) The Hearer interprets this combination of linguistic and extra-linguistic information produced by the speaker to be a reference to some particular object.
6) The game succeeds if the Hearer correctly interprets the information provided by the Speaker; if the agents do not agree on the object referenced, then the game fails.

Presumably, upon completion of a *naming game* interaction, learning occurs; while the bulk of research which follows this paradigm uses some variation of the adaptive rules outlined in [9], there is no reason to suppose that this framework must be coupled with that particular set of learning rules. In fact, we demonstrate that the *naming game* provides an excellent paradigm for use with agents possessing different internal mechanisms and learning procedures.

The procedure outlined above is the *naming game* in its simplest incarnation; many enhancements can, and have been, made, including noisy channels for both linguistic and 'visual' communication, and changing populations of both agents and objects.

One might imagine that the *naming game* is not a valid model of language acquisition, since the Hearer has no way of knowing that the spoken utterance names the object; it could conceivably be any form of communication, or even none at all. However, the *naming game* is intended to model neither language acquisition nor the origins of language, but rather language coalescence and change through highly constrained interactions between adult speakers.

The Phonetic Model

Rather than retain the character-based randomly-generated words of earlier systems, we have chosen to move towards linguistic validity by the inclusion of a rudimentary feature-based phonetic model.

Our agents communicate by means of single-syllable utterances consisting of a consonant followed by a vowel. Each phoneme is represented by a set of binary features loosely based on Chomsky-Halle features and the cardinal vowel system (see Table 1)

Table 1: Binary Feature Matrix for Phonemes

<table>
<tr><td></td><td colspan="16">Consonants</td></tr>
<tr><td>IPA</td><td>p</td><td>b</td><td>f</td><td>v</td><td>t</td><td>d</td><td>s</td><td>z</td><td>c</td><td>ɟ</td><td>ʃ</td><td>ʒ</td><td>k</td><td>g</td><td>x</td><td>ɣ</td></tr>
<tr><td>Ant</td><td colspan="4">+</td><td colspan="4">+</td><td colspan="4"></td><td colspan="4"></td></tr>
<tr><td>Cor</td><td colspan="4"></td><td colspan="4">+</td><td colspan="4">+</td><td colspan="4"></td></tr>
<tr><td>Vcd</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td></tr>
<tr><td>Cont</td><td>-</td><td>-</td><td>+</td><td>+</td><td>-</td><td>-</td><td>+</td><td>+</td><td>-</td><td>-</td><td>+</td><td>+</td><td>-</td><td>-</td><td>+</td><td>+</td></tr>
</table>

<table>
<tr><td colspan="8">Vowels</td><td></td></tr>
<tr><td>i</td><td>u</td><td>e</td><td>o</td><td>ɛ</td><td>ʌ</td><td>a</td><td>ɑ</td><td>IPA</td></tr>
<tr><td colspan="2">+</td><td colspan="2">+</td><td colspan="2"></td><td colspan="2">-</td><td>Closed</td></tr>
<tr><td colspan="2">-</td><td colspan="2">+</td><td colspan="2">+</td><td colspan="2"></td><td>Mid</td></tr>
<tr><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>-</td><td>+</td><td>Back</td></tr>
</table>

There appears to be little to differentiate a model consisting of a sequence of characters from one which consists of a sequence of abstract phonemes represented by binary feature sets, but the phonetic model we introduce does in fact provide at least one major advantage other than the semblance of cognitive validity. Rather than forcing each of the features to have a discrete binary value of either zero or one, we allow values across the real interval (0,1). Not only does this allow a much more flexible connectionist implementation than the equivalent using binary features, but it is in fact phonetically justified. The cardinal vowel system is little more than a set of standard reference locations for the infinitely variable tongue position observed in vowel production in the real world. Similarly, voicing delays on consonants vary from speaker to speaker and context to context, as do tongue positions in consonant.

By moving away from the character-based model, which can only encode a fixed amount of information depending on the character set, we arrive at a representation which allows us to encode a much higher degree of variability in the utterances, modelling crudely the acoustic signals received by the human ear.

Agents' Internal Neural Nets

Each agent is furnished with two completely separate neural networks, one for determining the utterance from precise object information (henceforth the S-Net, or Speech Network), and one for settling upon a particular object given

---

[1] Usually these are software agents and virtual objects, but some work has been done with autonomous robots. [9]

an utterance and (possibly) some non-linguistic information (the H-Net, or Hearing Network).

The S-Net

The neural net used for the production of utterances is of extraordinarily simple design. It is a two-layer, fully connected network, with an input node for each distinct object in the simulation and one node in the output layer for each phonetic feature in a word. Once the Speaker has randomly chosen an object as the topic, the activation of the corresponding input node is set to 1.0, while that of all other nodes becomes 0.0. The activation level on an output feature node is simply the sum of all inputs to the node, with no use of a threshold or normalization. Thus, the weights on the links from the active object node appear directly on the output layer; the range of values for the weights is the continuous interval [0,1], as this also defines the values desired for our features. The activations of the output nodes can therefore be interpreted directly as values for the corresponding features.

The H-Net

The H-Net is also a two-layer, fully connected network, with an output layer which is virtually competitive; inhibitory links are not implemented directly, but rather through a winner-take-all choice.

The output of the Speaker's S-Net is placed directly on nodes in the input layer of the Hearer's H-Net, modelling the reception of auditory information. There is a further subset of the H-Net input layer which is dedicated to extra-linguistic information.

This extra-linguistic information is meant to correspond vaguely to the real-world visual cues experienced by the Hearer in a *naming game* where the Speaker points at an object. Accordingly, the topic receives the highest score of all objects: not a perfect score, but rather a random number between ⅔ and 1. Four or five other potential objects are assigned smaller scores between 0 and ½ to represent physical proximity to the topic, the main source of ambiguity in pointing. The random choice of error-objects and the high degree of variability in the object scores is an attempt to crudely model a wide variety of pointing situations, where the topic will be surrounded by different objects in different configurations in every interaction. This differs significantly from a fixed object layout, pointed to in every interaction, since in that restricted instance, certain objects will never need to be disambiguated by phonetic information.

Once the input layer is fully initialized, activation levels for the output layer are calculated with a straight sum of products rule. The scores of the object nodes are compared, and the node with the highest score is chosen as the eventual winner of the virtual competition within this layer.

This object chosen by the H-Net is compared with the original topic, and the success of the game determined.

Training Regimen

Individual speakers are unlikely to drastically change their speech patterns when they are being understood; similarly, if a listener is able to comprehend a speaker, there is little reason to adapt one's model of the language to their accent. Accordingly, in our model, learning only occurs when the *naming game* fails.

In the real world, communication occurs for a purpose, and in the case of a misunderstanding about the topic of a conversation, it is unlikely that the participants will simply give up; the speaker will repeat the word, and perhaps even identify the object physically in an unambiguous manner (i.e. by picking it up). It is therefore reasonable to suppose that the Hearer agents are familiar with both the utterance produced by the Speaker and with the intended topic, even when the *naming game* does not succeed. We have arbitrarily chosen to have the Hearer adapt its behaviour to match the Speaker; when discussing this object in the future, the Hearer's speech will more closely resemble that of the Speaker, and the Hearer will also be more accepting of utterances similar to the Speaker's designation for that topic.

Initialization

All phoneme-object weights in both the S-Net and the H-Net are initialized to random values between 0 and 1, representing in the first instance phoneme values, and in the latter relative contribution of features to the object score.

Since the weights in the H-Net between the input and output object nodes undergo no training, their initialization must be performed more carefully. Weights between input and output nodes which represent the same object are set to 0.6, while all other weights in this set are given random values uniformly distributed over the interval [0,0.5]. This approach attempts to model in a simple way similarities between objects, while avoiding the undesirable extremes where object information either overpowers the contribution of the object's name or cannot affect the result.

S-Net Training

One of the tasks of the Hearer is to interpret the continuous phonetic output of the Speaker in terms of idealized binary features. This is implicit in the normal actions of the H-Net, but explicit during S-Net training; rather than adapting its speech towards the actual output of the Speaker, the Hearer moves its speech towards an idealized binary feature set. Because we only train the Hearer when the *naming game* fails, its speech will never reach this ideal, but will only move in that direction as far as is necessary for effective communication to occur.

$$w' = \sqrt[3]{\frac{w - 1/2}{4}} + \frac{1}{2} \tag{1}$$

$$w' = 4(w - 1/2)^3 + 1/2 \tag{2}$$

Each feature in the S-Net of the Hearer is examined independently to determine if its idealized value is the same as that of the corresponding feature in the Speaker's utterance. If so, its value is reinforced (see equation 1); if not, it is punished (see equation 2). The punishment equation moves values towards 0.5, while the reinforcement function moves values towards (but not beyond) 1 or 0, depending on the polarity of the weight. A random number between -0.05w' and 0.05w' is then generated and added to this new value, and this 'fuzzy' result is forced within the interval [0,1]. This last step is required, else punishment will set the weight on an exponential growth pattern. The random fuzz is also necessary, since the punishment function, on its own, will never force a weight across the fixed point of 0.5.

H-Net Training

The only weights in the H-Net which are trained are those between the phonetic input nodes and the output nodes (as discussed above, the object weights remain fixed at their initial values). Again, training only occurs when the Hearer has chosen the wrong object; the goal of this weight modification is to make the H-Net more likely to settle on the correct topic when given similar phonetic input in the future.

There are a number of ways to accomplish this result, but we settled on decreasing the score of the false positive, and increasing the score of the correct answer. This is a very straightforward procedure which does not overly complicate the dynamics of the network, and tends to restrict the weights to a reasonable range.

$$w' = w - \delta \cdot w \cdot p \tag{3}$$

$$w' = w + \delta \cdot w \cdot p \tag{4}$$

At the implementation level, we apply equation (3) to the false positives, and equation (4) to the missed answer; in these equations, w' is the new weight, w is the old weight, δ is the learning rate, and p is the value on the phonetic input node. Essentially, we modify the weight by a certain percentage of its contribution to the activation of the object node in question. The current value of δ in the system is 0.05, and since the value of p ranges from 0 to 1, in practice, the weight is modified by an average of 2.5% of its own value.

The Simulation World

We have tested and run our simulation with up to 50 agents and 20 objects, but for the most part we have kept to 20 agents and 10 objects, so that our results are comparable to those of [9]; these numbers seem to produce interesting results, yet have simulation run times which are reasonable.

Each agent has its H-Net and S-Net randomly initialized as described above. There is no internal communication between the nets, and we do not explicitly train the agents to understand their own utterances; the eventual consistency exhibited by the system is a result of self-organization (at a societal, rather than agent level).

We conduct instances of the *naming game* in groups of 20; the agents speak in order, and a random partner is chosen to be the Hearer. This approach has the advantage that speech starvation will not occur – every agent gets its chance to speak – but it is theoretically possible that an agent could survive a simulation completely unchanged, never being selected as the Hearer. However, the probabilities involved are so small that it is not an issue at present, and starvation-avoidance techniques could be easily added it if became a problem.

In the next section, we present results from four different types of simulation runs: with and without population flux, with either 5000 or 20,000 groups of *naming game* interactions. Since each group consists of 20 *naming games*, altogether the simulations consist of 100,000 and 400,000 instances of the *naming game*. In the simulations with population flux, a random individual is removed every 2000 games and a new, randomly initialized agent takes its place; in the longest simulations with population flux, there have been 200 new individuals inserted in the population.

Experimental Results

In some initial experiments with a learning rate (δ) of 0.20, we achieved an 86% average naming game success rate in interactions without object information given to the Hearers, and a 95% average success rate when such information was provided. This latter figure rose to 98% when the learning rate was changed to 0.05. We recently ran several simulations using only the object information, which resulted in a success rate of around 50%.

In our long-term trials, we achieved success rates around 95% when the population was stable. Certain periods of the simulation exhibited success rates around 98%, but the global average was lower because of language change and periods of instability. With a dynamic population, the success rate hovers around 80%.

Form Distributions

In order to quantify the development of our agents we plotted, for each object, the number of speakers for each linguistic variant as a function of time. Since we ran 15 simulations with a static population and had 10 objects in each, this gives a total of 150 separate graphs over the length of 100,000 games, 60 over 400,000 games. Examining these graphs, we recognized that they fell into several different patterns, based both on the overall appearance of the graph, and some underlying statistics.

In *dominance*, one form (almost) completely dominates the phonetic space quickly, and retains control for the duration of the simulation (See Fig 1a). In *70-30* graphs, two forms exist in the population, one spoken by about 70% of agents, the other by about 30% (See Fig 1b). *Parity* graphs have two common forms, splitting the majority of speakers between them (Fig 1c). When a large number of competing forms arise (usually 4), none spoken by more than 40% of the population (See Fig 1d), we call this a *mush* graph. In the *step-up* pattern, one form appears destined for dominance, but it is overtaken

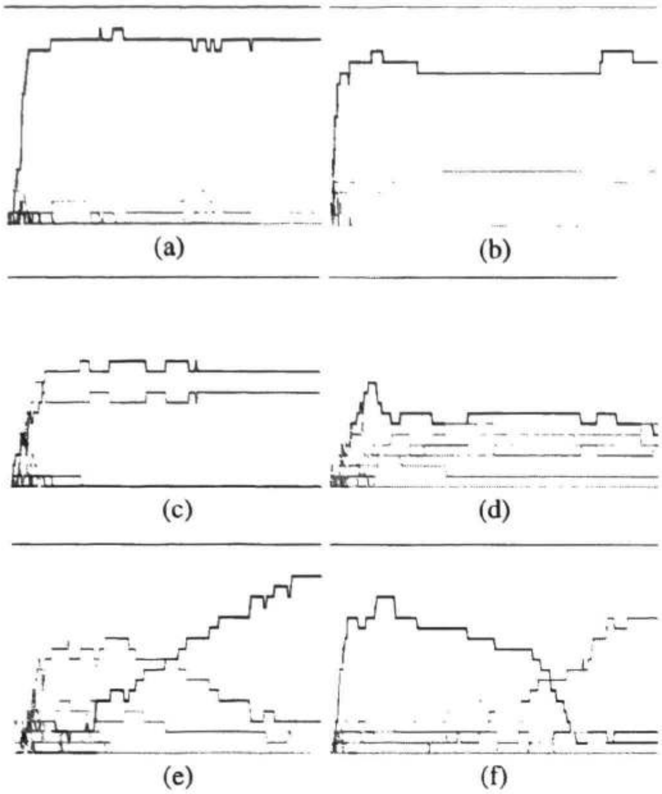

Figure 1: The graphs above represent the number of speakers of linguistic variants as a function of time for a particular object. The vertical axis counts the number of agents speaking a form, while the horizontal axis represents time. Each shade of gray shows the frequency fluctuations over time for a particular linguistic form. Each graph is a canonical example for its category: (a) dominance (b) 70-30 (c) parity (d) mush (e) step-up (f) switch. As an example, graph (c) shows a simulation where two forms, after an initial learning period, achieve a steady state where each is spoken by about half the population.

Table 2: Relative Frequency of Form Patterns

Form Patterns	% of Static Pop.		% of Dynamic Pop.	
	100k games	400k games	100k games	400k games
Dom.	30	38	30	21
70-30	9	17	9	2
Parity	16	12	9	3
Mush	30	7	12	3
Step-Up	9	17	26	24
Switch	6	10	13	47
Total # of Graphs	150	60	160	70

by a second form, which goes on to dominance (See Fig 1e). Finally, some of the graphs exhibit a form *switch*, where one form dominates with over 60% of the speakers for a period, and then is replaced by a different form, which then dominates (see Fig 1f).

Table 2 outlines the relative frequency (as a percentage) for each category of graph. We show statistics for both our short and long simulations, separated into runs with and without population flux.

Obviously, these categories have very fuzzy boundaries, and some graphs simply do not fit particularly well into any category. However, there are clear examples of each group, (including the *switch* form representing lexicon change) and the fact that these particular patterns are the most common provides insight into the learning processes of our agents.

Discussion

We are not willing to claim that coherence in language **must** be due to self-organization, but our simulation (along with [8] and [9]) makes it clear that extremely simple self-organizing systems **can** achieve a coherent lexicon. Our results reinforce the idea (presented in [9]) that population flux increases the incidence of lexicon change, but we also show significant change even in a static population (see [3] for a discussion).

When we examined the graphs of our simulations, we at first had a difficult time reconciling a 95% success rate with the fairly high frequency (30%) in short simulations of the *mush* graph, where there were multiple competing forms. However, an examination of the lexical forms in these mush patterns revealed that all forms were fairly similar. The combination of these forms and the weights on the network showed that our H-Nets were acquiring schemata, which may include one or more □ (don't care) values, to use the notation of [5]. For example, a particular H-Net recognized a voiced anterior consonant followed by an 'i' as Object 8, regardless of whether the input utterance was 'di', 'zi', 'vi', or 'bi'; the agent has internalized a schema which includes a □ value for the coronal and continuant features.

Schemata allow us to explain not only the high success rate in the face of many variants, but also the stability shown by the variants themselves. No matter which of the four forms above is heard, the H-Net will settle on Object 8; this allows all four variants to flourish, since in the absence of misunderstanding, no learning occurs, and the forms are stable.

With our schemata firmly in hand, we can also investigate the high degree of stability demonstrated in the *parity* and *70-30* graphs. Both patterns exhibit two forms which together dominate the population of speakers, which can be nicely explained by the acquisition of a single □ value by the H-Nets of the population. The two patterns in fact represent different facets of the same underlying situation: in the *parity* case, the single-□ schema first dominates the population at a time when the two forms have approximately equal shares of the speakers, whereas in the *70-30* pattern, this stability is achieved when the distribution is somewhat lopsided. The *dominance* graphs, of course, are a result of H-Nets learning fully specified schemata (i.e. no □ values).

Even the *step-up* graphs rely to some extent on the existence of schemata in the population of H-Nets. In this pattern, one form achieves a certain degree of initial prominence amongst the speakers, but is quickly overtaken by a similar form. In fact, this occurs when part of the H-Net population settles on a schema having a □ value which allows the initial form. If at some point before this schema dominates the population (which would result in parity), a significant portion of the H-Nets learn a fully specified schema for a different form which is also allowed by the □-schema, this new form will eventually take over. Even if the usurper's frequency is initially low, the new form will dominate, as it is meaningful to all agents, whereas the first form is understood only by those with the □-schema.

Another interesting phenomenon is the near-disappearance of the *mush* pattern in longer simulations, especially those involving population change. This is a more drastic example of the process explained in the previous paragraph. Longer simulation runs involve more lexicon change, and this tendency is exacerbated by a changing population. The phonetic space in our model is limited; since a *mush* form occupies far more than its share of this limited resource, there is a certain degree of pressure to reduce the number of □ features. If for some reason a form for another object moves into this space, object-confusion will result. This will apply pressure to differentiate the two schemata; the simplest adaptation is simply for the *mush* pattern to lose one of its □ values, resulting in a *70-30* or a *parity*. Of course, this same process is moving these single-□ patterns into fully specified *dominance*.

Although less probable, the reverse operation also occurs in our simulation, with *parity* patterns (one □) becoming *mush* patterns (two □s). Over very long simulations, one would expect these two forces to come into balance, resulting in a relatively stable distribution of □ features over the population.

Conclusions and Future Work

While the results reported in [8] and [9] are exciting, the authors make little attempt to exhibit any sort of low-level cognitive validity. Our approach recasts this earlier work in a more natural form, introducing a connectionist cognitive model for the agents and a much richer phonetic model. We have also refigured the *naming game* paradigm as implementation-independent, divorcing its description from the details of the accompanying model, a characteristic which is distinctly lacking in other definitions.

Our most significant result is not merely that language emerges from our system, which we've taken care to provide with a cognitively valid base, but rather that the linguistic systems which our agents learn are themselves cognitively valid. The schemata learned by the agents do not just provide an explanation of their behaviour, but represent valid phonetic generalizations in their own right. Human speakers do not learn fully specified feature sets, but rather schemata with one or more □ values. For example, English speakers do not differentiate phonemically between aspirated and non-aspirated consonants, whereas this has been constructed as a distinctive difference in proto-Indo-European, a distant ancestor.

Clearly, some of the most exciting future work involves following up the notion of the schemata which our agents learn, determining how the distribution of schemata affects the evolution of the system, and to what degree future behaviour of the system can be predicted. These schemata should also prove crucial in planned investigations of the complex interactions at the border of two stable languages.

Both the phonetic model and the object model used in our simulation could be improved. We plan to model physical constraints of the vocal tract so as to have the agents produce even more realistic sound combinations, which will allow us to expand the feature set and thus the number of phonemes. We hope to introduce an object model where objects are represented by feature vectors rather than simply atomic nodes, to see if hierarchical concepts might be instantiated as lexical items under these conditions.

Our results build on other recent work, demonstrating not only that modelling language as the emergent behaviour of a complex adaptive system can be a valuable tool for linguistic investigation, but that these systems can be created in a cognitively valid manner.

References

[1] Carr, Philip. *Phonology*. London: The Macmillan Press, Ltd, 1993.

[2] Caudill, Maureen and Charles Butler. *Naturally Intelligent Systems*. Cambridge, MA: The MIT Press, 1991.

[3] Dircks, Christopher, and Stoness, Scott C. "Lexicon Change in the Absence of Population Flux", Proceedings from ECAL 99, Springer-Verlag, forthcoming.

[4] Gurney, Kevin. *An Introduction to Neural Networks*. London: UCL Press Ltd., 1997.

[5] Holland, John H. *Adaptation in Natural and Artificial Systems*. Cambridge: MIT Press, 1995.

[6] Rogers, Henry. *Theoretical and Practical Phonetics*. Toronto: Copp Clark Pitman, Ltd., 1991.

[7] Steels, Luc. "Self-organizing vocabularies" In: Langton, C. (ed.) (1996) *Proceedings of Alife V*, Nara Japan.

[8] Steels, L. and F. Kaplan. "Stochasticity as a source of innovation in language games," *Proceedings of Artificial Life VI*, Adami et al eds. Los Angeles: MIT Press, 1998.

[9] Steels, L. and F. Kaplan. "Spontaneous Lexicon Change," *Proceedings from COLING-ACL 98*. Montreal: Université de Montréal, 1998.

[10] Steels, L and P. Vogt. "Grounding adaptive language games in robotic agents." In Harvey, I. et.al. (Eds.) *Proceedings of ECAL 97*, Brighton UK, July 1997. The MIT Press, Cambridge Ma., 1997.