

# A Workbench for Simulating Natural Language Evolution

Carl Vogel and Justin Woods

Carl Vogel is a faculty member in the Centre for Computing and Language Studies and the Department of Computer Science at Trinity College, University of Dublin. Justin Woods is a graduate student in the Computational Linguistics Lab at the same university. (email: [vogel@tcd.ie](mailto:vogel@tcd.ie))

### Abstract

We present an implemented architecture for simulating the evolution of natural language. The workbench allows parameterization of a large number of assumptions that researchers in the field work with, allowing interactions of parameters to be explored in detail. The system is based on a social rather than genetic model of language evolution, on the basis that complex interactive structures like stock exchanges are likely to lack genetic encoding. Rather, the items selected for continued existence in this system are mappings between phoneme sequences and underlying meanings. Representative experiments enabled by the system are discussed.

## I. INTRODUCTION

In recent years there has been a resurgence of interest in language evolution, with a great deal of work in computer simulations of evolution dynamics [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. This paper presents an implemented workbench for experimenting with simulated evolution of language in which a range of interesting parameters may be explored. In the sense of [11], ours is a system for parameter *tuning* rather than automatic adaptive parameter control, because language evolution research is at the stage at which parameter individuation and extensive experimentation are most appropriate. The adaptive control scenarios discussed by [11] are best suited to mature evolutionary models, after consensus has emerged about what the relevant parameters are, separate from whether their settings are static or dynamically adjusted. Recently, tools for monitoring the evolutionary dynamics of other systems have been introduced [12], [13]; however, the system presented here is tailored to the specific sorts of research questions that are current in the literature on evolution of language.

The system that we present is based on a social model of language evolution. That is, we do not presume that selective advantage accruing to language users force a genetic encoding of linguistic abilities. Encoding of linguistic functions in the genome is consistent with our approach but not required by it. We take, instead, the perspective that the relevant location for selection at the start of language use and social transmission is in mappings between phoneme sequences and meanings. Thus, we have a system for exploring parameters that impact on a social model of evolution. Economic exchange mechanisms are analogously complex systems of interaction whose underlying nature are inherently social rather than being encoded in the human genome. Our rationale for pursuing research with such a weak assumption is that even

if language is heavily encoded in the genome through selective processes, at the very start of language it must prove socially effective, and it must do so very quickly — if it takes thousands of utterances for the first language users to have successful communications, then it can be expected that language will not last as a cultural artifact, let alone as a source of functionality that provides speakers selective advantages.

The workbench has been used to explore a range of parameters that may have independent or interactive impact on the potential for language evolution [14]. The system can be used to study the role of the size of the space of available phonemic distinctions, the size of the space of entities discriminated (or effectively infinite discrimination with variables rather than labels), the space of discriminated event types, the number of conversation games embarked upon, the duration of a conversation game sequence, a measure of attention span in terms of the number of times a pairing of phonemes with meanings must be repeated before being encoded in long term memory, absolute minimum levels of communicative success, probability in task based settings that any individual communicative act must be successful and if it must be successful, the minimum level of success necessary, sensitivity to recency versus frequency effects, random distribution of the meaning space or Zipfian distribution of event types. This article describes the system, implemented in Prolog, through its primary data structures and algorithms. The sections that follow justify the parameters that are provided and indicate how the data structures and algorithms are sensitive to acceptable value ranges for those parameters. We do not here embark on systematic exploration of interactions among the variables, a separate thread of our research, but illustrate the sorts of experiments that may be conducted with the system. We also indicate the nature of our most immediate plans for extending the system.

## II. PARAMETERS AND PROBES

The experiments sketched in §IV and those conducted by [14], [15], [16] are based on a general architecture for simulating social models of language evolution. The following set of parameters is provided within the system; these are outlined in greater detail in §II-A. In a number of cases (especially relating to levels of feedback and degrees of assumed understanding) the system allows parameter tuning where past systems have proceeded from hard-coded assumptions.

- Size of phoneme space

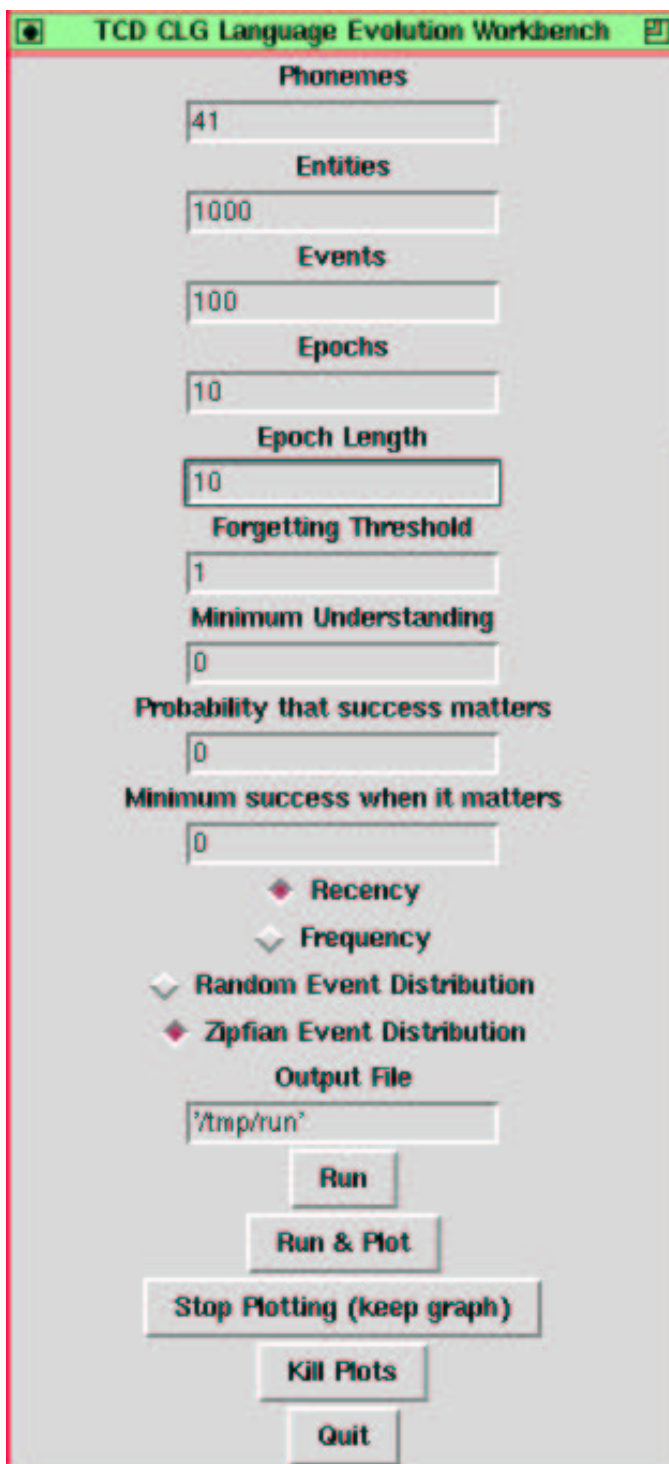


Fig. 1. GUI for Parameter Setting

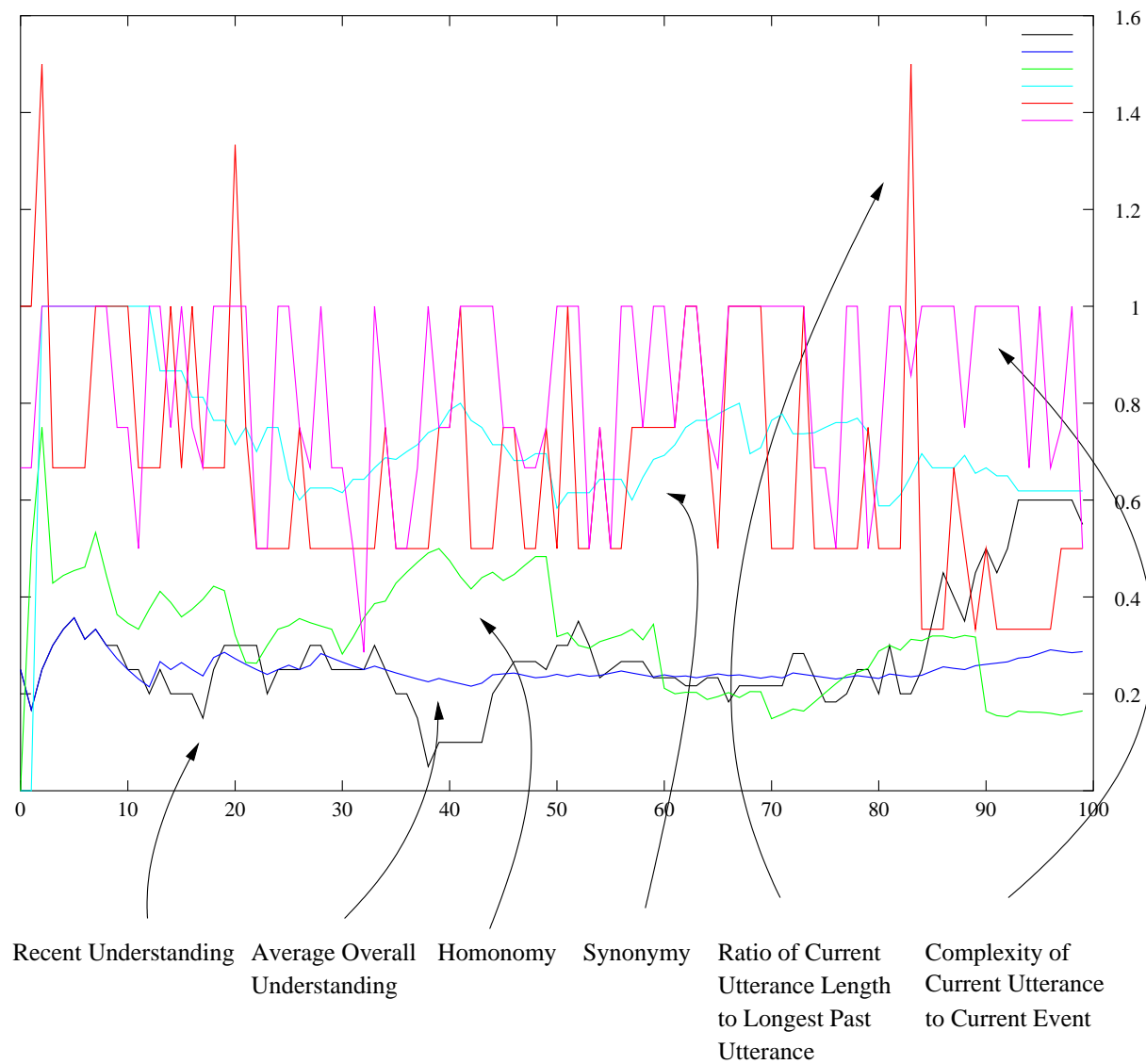


Fig. 2. Relevant Measures for a Run with Parameters Set as in Fig.1

- Size of entity pool
- Size of relation pool
- Random vs. Zipfian distribution of relation types
- Number of epochs
- Attention span
- Forgetting threshold
- Minimum required level of understanding
- Probability that success matters

- Level of necessary success
- Sensitivity to recency vs. frequency

Figure 1 displays the TCL/TK user interface for specifying these parameters. Experiments in tuning the parameters in the context of the system are reported in [14], [15], [16].

In the remainder of this section we provide an indication of the rationale for parameters individuated (II-A) and the sorts of measures that are made to calculate the degree of convergence within the system (II-B). On the latter issue, the features of communicative histories that we monitor are the following:

- Level of understanding achieved in a communication.
- Average understanding of last 10 utterances
- Average understanding overall.
- Number of words uttered
- Number of word types
- Homonymy ratio
- Synonymy ratio

Figure 2 provides graphical output over time of a number of these measures, using the input parameters as specified in Figure 1. The y-axis ranges at intervals of 0.2 between 0 and 1.6 indicating ratios measured at each step in conversation given by the x-axis.

### A. *Specific Parameters*

In this section we provide more details about the parameters that may be explored in our system. We describe here how the quantities parameterized are represented for the algorithms to process in greater detail in §III-A.

1) *Size of phoneme space:* A phoneme is modeled as a Prolog atom; arbitrary numbers of atoms can be constructed using the built-in relation `name/2`. Speakers pair phonemes and model words as lists of phoneme pairs. Clearly there is a relationship between the number of phonemes and the number of distinct words that can be expressed. By default (not via specifiable parameter) when a speaker innovates a word for a meaning the word is a pair of phonemes. The pair is arbitrary—nothing constrains the tuple of phonemes in terms of possible articulation, nor is there a requirement of analogical mapping for re-use of any expressions. Thus, innovations may be consistent with analogy to prior mappings, however analogy is not assumed as

a mechanism that fundamentally drives innovation. Because the choice of ways to express a meaning is partly determined by past experience, and because speaker at times acts as hearer, then because hearers may segment sound streams differently words may have arbitrary lengths (e.g. if sound streams for complex events get interpreted as referring to single participants within those events). The size of the available phoneme space is one of the parameters whose range of settings explored by [14], [15], [16].

2) *Size of entity pool:* The size of the entity pool has divergent dimensions. There are three entity types existing in a hierarchy. Either non-negative integer supplied and that number of arbitrary Prolog atoms are generated and randomly assigned among those types, or the atom `vars` is supplied. In the latter case we can explore the proposal [3] that in protolanguage there are no proper names, only predications over discourse referents [15]. When that is the setting, there is only one variable of each sort in the fringe of the hierarchy of entity-types. Predications then are made over variables without assignment functions anchoring variables into the world (of course, assignment functions would reinstall the behavior of proper names). Because we do not yet facilitate discourse relations across sentences, there is no other palpable difference in using entity names vs. variables.

Obviously the values we measure for utterance types, homonymy and synonymy will depend not just on the number of distinct phonemic forms mapped to meanings, but also on the total number of things that can be discriminated as individuals. This can be pursued in two distinct ways. One specifies a finite bound on the number of individuals discriminated for conversation, the other, using variables, effectively allows an infinite number of individuals to be referred to but constrains the number of sorts to three.

3) *Size of relation pool:* There is a fixed range of possible event types, abstracting over event names and names of selected arguments.

- 1) Arity 1, Animate argument
- 2) Arity 2, Animate arguments
- 3) Arity 3, Human arg1, unconstrained arg2, Animate arg3
- 4) Arity 3, Animate arg1, inanimate arg2, animate arg3
- 5) Arity 2, Human arg1, Relation arg2 (recursive)
- 6) Arity 1, unconstrained argument

7) Arity 3, Human arg1, unconstrained arg2, Relation arg3

The size of the relation pool specifies not the number of event types in the above sense, but the number of event names distributed over those types. This is distinct yet again from the number of event particulars that may occur fixing an event name with appropriate arguments at particular moments in time. The event types differ in their arity and in the constraints placed on the types of their arguments. Arguments can be animate or inanimate entities, animate entities potentially further classified as human. Arguments can also be relations as well. Because an event type can embed an event (as in ‘Leslie believes the boat arrived’), there can be arbitrarily many distinct events, without a finite bound on the size of the space of possible meanings. No tense or aspectual information is encoded. Further, there is no discrimination of negation as an event operator (thus, anthropomorphizing, there is only ‘seeing’ and ‘overlooking’; there is no ‘not seeing’ — in broad strokes, this is compatible with the assumptions in the Mental Models literature about initial cognitive representations of information lacking negative information [17]).

4) *Random vs. Zipfian Distribution of Event Types:* At initialization, based on the number supplied for the parameter of the relation pool size, a number of arbitrary relation names is constructed as Prolog atoms and assigned to one of the seven possible event types. The distribution across those types is random. Depending on a binary choice for a parameter related to this distribution, the consequent frequency of events with a particular relation name is either utterly random, or based on a Zipfian distribution with respect to the number of events entered as a parameter in which event frequency is inversely proportional to rank. In the latter case probabilities of generation are assigned to each event name according to the Zipfian Distribution. According to the Zipfian distribution, events with particular relation names will occur with different frequencies to others—ranking those occurrences by frequency, the highest ranked event name will be roughly twice as frequent as the next in rank, three times as frequent as the third in rank, and so on. The actual relationships are determined by the number of individual event names. This weighting of events occurs without generating any events anchored to particular arguments, so infinity of semantic space is preserved. The intuition behind allowing the parameter is that observed Zipfian distributions of linguistic types may partly result from corresponding systematicity in underlying reality that linguistic types tend to be about. [16] reports in detail on experiments involving this parameter.

The probability of the top ranked event  $p(x_1)$  is calculated as follows:

$$p(x_1) = \varphi \left( \sum_{k=1}^{\varphi} \frac{\varphi}{k} \right)^{-1}$$

where  $\varphi$  is the total number of events to be generated and  $k$  is event rank.

The probabilities of the remaining events are calculated as a function of  $p(x_1)$ :

$$p(x_r) = \frac{p(x_1)}{r}$$

A full description can be found in [16].

In any case, any given run of the system need not have an instance of a relation name for each of the possible event types. If no such relation name exists after initialization, that omitted possible event type will never occur. Similarly, there may be many different relations of each possible sort. Each relation name for an event type counts itself as an event type, distinct from an event (an occurrence of an event type). This is just the same as a human language having many different verbs that count as transitive, and many instances in the world of situations described by the event types that correspond to the verbs.

When small numbers of entity types are discriminable and larger numbers of relation types are available, then there are chances that a potential event type is constructed which can never be realized because no entity exists which satisfies its argument constraints.

An event happening, as in the instigation of a speaker's utterance is modeled by random selection of a relation name (hence an event type), and random selection of elements for the arguments, subject to the constraint that the arguments have the right semantic category and subject to the Zipfian distribution constraint if that parameter is set. There is no in-built proclivity for any one sort of event type over any other (when the Zipfian distribution is not used) beyond the fixing of the seven abstract sorts, and random distribution of the parametrically specified number of relation names over those sorts. By construction, the number of possible distinct combinations of event with arguments is infinite.

We see it as extremely important that the space of possible things to talk about not be by assumption finite. Note that some research assumes finite meaning spaces [18]; similarly, the architecture of [1] depends on finite association matrix manipulations and hence finite bounds on the number of meanings. [19] provides simulations in which compositionality emerges within a

system about which 100 meanings are available. In our system there is no guarantee that even with extremely long runs that an event will recur with identical arguments to the relation. It is not possible to ensure ultimate success by depending on repeated associations to a finite number of possible meanings.

5) *Iterations & Forgetting*: Two parameters to the system determine the number of conversations attempted. This is in conjunction with another parameter that relates to how frequently associations have to occur before they are remembered for re-use. Thus, the **Attention span** is the maximum number of utterances that an agent can sustain before forgetting any associations which have occurred a number of times less than or equal to the number provided as the **forgetting threshold** among everything so far uttered. The number of utterances given as the attention span is one epoch, and the **Number of epochs** determines how many epochs will be run. These parameters jointly determine a lower bound on the number of words and word types that will be constructed during a run. When utterances are made, they are remembered (that is, unless they fall on the wrong side of the forgetting threshold at the end of an epoch). When speaking about something new, the speaker may take into account past speakings or past interpretations, and similarly hearers. It is reasonable to examine different assumptions about interlocutors' attention spans in remembering past associations. If a word type, an association between a phoneme sequence and a meaning, occurs only once in the course of 100 conversations it could easily be forgotten. The value set for the forgetting threshold determines the minimum frequency of required repetition.

6) *Minimum Understanding Threshold*: One might want to assume that the first utterances of human languages were guaranteed some minimum level of understanding [1], [20]. An optimist might even claim that the language constructed from the very first utterances benefitted from perfect understanding of those utterances. Our own view is extremely negative on this point. However, the system nonetheless allows this assumption to be parameterized. The effects are interesting. Extreme pessimism about the amount interlocutors inventing language for the first time actually understand does not doom the system to failure; under reasonable parameter settings, even setting requisite understanding at 50% can lead to unrealistically high levels of convergence in understanding quite quickly.

7) *Feedback*: In our system it is possible to parameterize two related values which jointly determine the sort of feedback available to a speaker. Recall that we specifically do not assume

that learners are presented with the exact meaning of the utterance from which they are trying to bootstrap language. A weaker assumption, though, motivated by task-oriented dialog is that the dialog situation may or may not provide feedback, particularly through the relative success of a communicative act. Thus, one value that may be specified is the **Probability that success matters** in any act communication. If the value is 40%, then there is a 40% chance that the particular event being talked about has any degree of urgency. Suppose some particular event lands in the 40% category, then the **Level of necessary success** determines the amount of the utterance that must be understood by the hearer for it to even register as having been an utterance. If it is not registered then it will not be available in the knowledge base for re-use. The level of understanding is measured as the percentage of associations between phonemes and meanings that are identical for both speaker and hearer with reference to the utterance at hand. Clearly, the most pessimistic situation involves zero for both values as if language can emerge there it emerges without recourse to transparent meanings and, in fact, without any stipulation that communication needs to be successful as a prerequisite for language to develop. This is the position adopted by [6] in which the communicative system is never measured against objective function so in effect the agents do not care if they communicate or not. In contrast, the system of [21] assumes that feedback is essential in all cases where communication fails. While we do not deny that active participation in tasks and task associated dialogs can generate feedback that is demonstrably useful [22], one cannot simply assume this mechanism at the very start of language's coming into existence any more than one can assume that parental corrections supply sufficient negative evidence for child language acquisition to avoid the unlearnability results for grammar induction [23].

8) *Sensitivity to recency vs. frequency*: A final experimental parameter<sup>1</sup> is set either as **r** for recency or **f** for frequency. This accommodates two possible ways that agents might be configured when deciding how to re-associate phoneme sequences with meanings. A recency based agent will tend to just re-use the last association, while a frequency based agent will be strongly influenced (but as noted above, not *determined*) by the greatest frequency of past associations of phonemes with meanings. An agent who is sensitive to recency rather than frequency has access only to local coordination processes as only the most recent prior event

<sup>1</sup>One other parameter is just the output file which records various probes in sequence for the omniscients. This file is given as input to gnuplot to produce the graphs included herein.

can have an impact on the current one, while frequency based agents have additional access to global co-ordination [24].

### *B. Specific Probes*

Figure 2 demonstrates an output of the system measuring a number of values through the course of execution. We emphasize that these values are visible to us as omniscient outside observers, but because of the parameter settings used (including zero feedback and zero prerequisite understanding with each speech act), the interlocutors behavior is not shaped by those measures of the system's dynamics. Here, as generally, we measured recent understanding as the average understanding by utterance over preceding 10 utterances. The understanding level of a particular utterance is computed as the percentage of the total phoneme–meaning pairs that the speaker and hearer agreed upon (even though they did not know themselves whether they achieved agreement). The average overall understanding averages this over all communications and not just the most recent 10. Homonymy is the percentage of phoneme sequences that point to more than one meaning, and synonymy is the percentage of meanings that have more than one name. Recall that natural languages, at the lexical level, tend to tolerate homonymy more than synonymy. The ratio of the current utterance length to the longest utterance to date gives a picture of average utterance complexity. Any time this measure exceeds the value 1, the utterance length of the current utterance has exceeded the length of the longest preceding utterance. The final quantity graphed here is a measure of the complexity of the utterance relative to the complexity of the event—the amount of the meaning segmentation that has its own segment in a sequence of phonemes.

Given that we're examining the creation of language, it is interesting to monitor the total number of words uttered as well as the total number of word-types. This is the normal type-token distinction, the total number of utterances vs. the total number of distinct pairings of phoneme sequences with meanings. Presumably, there is a relationship between the phoneme space and the ratio between types and tokens. That is, given a fixed number of utterances, a number of possible phonemes leads to a proportionate space of pairings between sequences of phonemes and meanings, hence a proportionately sized set of types and type-token ratio. However, other meaning based assessments of linguistic innovation are also necessary. Evidently, natural lan-

guage tolerates more homonymy than synonymy,<sup>2</sup> even though if one were rationally designing a language one would prefer the alternative relationship as enhancing the probability of being understood<sup>3</sup> given that homonymy increases the chances of being misunderstood. While we accept a point emphasized recently [18] to the effect that homonymy is most acceptable when the intended meanings of homonym sets is maximally distinct, we nonetheless take it as an independent validation of linguistic properties within a simulation system for the number of synonyms to be smaller than the number of homonyms.<sup>4</sup> In any case, our system does in fact allow for both homonymy and synonymy. Simulations in which the agents are presented with both words and meanings during the lexicon learning process do not allow certain observed phenomena of natural language such as polysemy or meaning evolution to arise [25].

### C. Discussion

We have thus far presented the background assumptions and architecture of our system for simulating language evolution. Richer environments are easy to imagine. One would like a system to be able to track information states of multiple dialog participants, and to allow questions and other forms of dialog moves in addition to declarative comments. In fact, [16] presents an architecture for dynamic agent information states and questions. Nonetheless, the architecture we provide does have a number of important features. The conceptual universe of things to talk about is essentially infinite, and inclusive of recursively constructed events. A range of parameters that have proven interesting within the literature to date are incorporated and have been described in detail above.

## III. REPRESENTATIONS AND ALGORITHMS

### A. Data Structures

While we do not assume that there is transparency of meaning, we do assume that agents have the same space of possible meanings available to them, and the same range of basic concepts.

<sup>2</sup>By ‘synonymy’, we mean a state in which one meaning can be denoted by a larger number of *basic* expressions. Natural language does support a larger amount of synonymy at the phrasal level.

<sup>3</sup>Presumably this is more urgent than learnability limits on the number of distinct ways one might have of referring to one meaning.

<sup>4</sup>We could, but have not yet, implement a mechanism for measuring distinctness of meanings of homonyms given the representations that we describe presently.

We also assume that agents exist in the same world of conceptual atoms. However, we do not assume that agents segment events into constituent events in the same way, even though we do assume that they decompose complex events into the same set of conceptual atoms. Entities, entity types and event types are represented as Prolog atoms, their numbers are determined by settings to input parameters. Events themselves are modeled as instances of Prolog lists pairing event types with arguments. Entities are classified as either human, animal or inanimate, with event types selecting arguments of type: human, animal, animate, inanimate, event type or without constraint.

In the current implementation, the following event types are possible:

- 1) Arity 1, Animate argument
- 2) Arity 2, Animate arguments
- 3) Arity 3, Human arg1, unconstrained arg2,<sup>5</sup> Animate arg3
- 4) Arity 3, Animate arg1, inanimate arg2, animate arg3
- 5) Arity 2, Human arg1, Relation arg2 (recursive)
- 6) Arity 1, unconstrained argument
- 7) Arity 3, Human arg1, unconstrained arg2, Relation arg3

Based on the numbers given as parameters, a corresponding number of event types and entities is created, randomly assigning them to the possible categories (unless a Zipfian event distribution is used; see below). Random Prolog atoms are constructed for each, varying in form for human readability. An example event is this: `[ihdixos, spmg, davr, fizg]`; where `ihdixos` was an atom constructed to correspond to an event type with an animate first argument, an inanimate second argument and an animate third argument.<sup>6</sup> Note that some event types embed relations as arguments. These are recursive events like seeing, or knowing in English in which an arbitrary event is embedded. Thus, although there are a finite number of events as provided by an input parameter, if one or more of those event types is of a sort that allows embedding, then during the course of simulation in which events happen and speakers comment on them, individual events may be arbitrarily complex. That is, in the limit there is an infinite number of possible events even given finite specification of the number of event types. This is distinct from

<sup>5</sup>An unconstrained argument may be any of the entity types or, recursively, an event type.

<sup>6</sup>Those were the properties of `spmg`, etc., in this particular run—nothing about that atom encodes animacy, and in another run it might have had a different type or not have been generated at all.

boundless iterability of a finite set of events and accompanying probabilities of repeated event types.

Perhaps more important is that the events do not themselves encode predicate argument structure in a way that that would prejudice learning towards the context free grammar implicit in a predicate argument representation of an event with the functor providing the left-hand nonterminal and the arguments supplying the right-hand side of a production. This is accommodated as follows (compare with [26]). When an event happens the speaker selects a random element of the set of all sublists of the event, each corresponding to a distinct perspective, an agent might have with respect to the example event.

An event happens: [ihdixos, spmg, davr, fizg]

Possible construal of the event: [[ihdixos], [spmg], [davr], [fizg]]

Possible construal of the event: [[ihdixos], [spmg, davr], [fizg]]

The hearer is not constrained to have the same perspective on the event as the speaker, grounding our approach in the perspectives of the interlocutors, and so the extent of shared meaning is dependent on shared perspective. This approach to the conceptual representation of agents is supported [27] by work which suggests that statistically good mappings will outnumber bad mappings and the system will learn. This is also consistent with the contention [25] that individuals should not all have the same linguistic competence.<sup>7</sup> In their experiments the conceptualizations and lexicons of the individual agents were *never* the same. Even if a shared perspective obtains the hearer will not have access to individual type-token relations, instead it will map the entire utterance to the entire event and then attempt to interpret the component lexical items and induce what any unknown items might mean.

Phonemes are also arbitrary atoms.<sup>8</sup> A speaker, upon witnessing some event, uses pairs of phonemes to refer to each element of the event. At the outset, this involves invented pairs for the association, but over time, past experience interacts. Thus, one element of constraining structure imposed on the model (not yet parameterized), is that speakers try to talk about the entirety of

<sup>7</sup>We assume only that agents probably had similar basic vocal and auditory capacity, but do not assume any shared linguistic or perceptive competence.

<sup>8</sup>Note that they are atoms and not letters—if an input parameter requires more phonemes than can be represented by arbitrary letters of the English alphabet, then atoms are constructed from longer sequences, yet in either case, as atoms they do not correspond to English letters.

their perspective on an event placed before them. However, through iteration between being a speaker and a hearer, a speaker may come to prefer a more complicated way of referring:

Event: [jilufks, furk]

Uttered: [[[jilufks], [r, q], [r, v]]], [[furk], [e, y]]]

Heard: [[vmpk], [r, q]], [[furk], [r, v], [e, y]]]

Here an event occurred, and a speaker used a four-phoneme complex, [r, q], [r, v] to pick out the event type and a two phoneme complex to pick out the argument. Thus, words are modeled as lists of pairs of phonemes and are not restricted in length. There is sufficient space for duality of patterning to emerge without its being built-in (see [28]). The phoneme pair is the minimum meaning bearing unit. If the two phonemes are identical, then phoneme meaning is the same as pair meaning, and there is no duality of patterning. The duality emerges because the minimum meaning bearing unit is constructed from arbitrary phonemes that hold no meaning on their own. Duality of patterning is built in to the extent that the chance that each phoneme will be paired with itself over a series of random selections is decreasing, in inverse relation to the size of the phoneme space, yet it remains a possibility for self-pairing. There is no model of articulatory constraints on possible phoneme pairings.

The hearer has unobscured access to the same event commented on by the speaker. This models the sharing of cognitive possibilities among communicating agents. However, perspectival divergence is also possible. The hearer can partition the event differently; thus, our model does not require agents to share perspectives. The onus on the hearer is not to find a phoneme sequence for each part of the perspective on the event, but something in the conceptual space for each phoneme sequence to mean. Another assumption that is not parameterized is that there is no noise — while the hearer may segment the signal differently (as in the example above), the hearer has perfect access to the stream of phonemes uttered. The lack of clarity is in what the utterance means. In the above exchange, although the hearer takes the speaker as relating the sole participant in the jointly witnessed event, the [furk], to some other entity associated with [r, q] in the past, and the entirety of this meaning is wrong, and although the segmentation is wrong leading the hearer to assume that a distinct signal is at stake ([r, v], [e, y]) vs [e, y]) at least the intended denotation is correct.

The asymmetry in responsibility between speaker and hearer does not contradict the Saus-

surean perspective. Speaker is required to comment on everything in its perspective on the event; hearer is required to ground every word it segments from the signal either in its perspective on the shared event or on past interpretations. Perspective is known to confound theoretical assumptions. Empirical research demonstrates the lack of speaker attention to hearer perspective in identifying use conditions for definites [29]. The asymmetry we suggest here contradicts the Saussurean perspective only to the extent that those empirical findings with human communicators do.

There is no initial grammar that constrains the system, neither explicitly, nor implicitly in the structure of semantic representations of events. There are only associations of meaning sequences with phoneme sequences (compare with [30], [26], [19]).

Clearly, although there are hard-coded assumptions like a lack of distortion in the speech signal, the responsibility of the speaker to comment on the entirety of an event and of the hearer to find a meaning for each part of an utterance, there is a lot of room for things to go wrong. Successful evolution of a useful language under realistically unfavorable conditions is more impressive than an evolutionary system with dice loaded for success. In sum, the basic representations involve structured lists of arbitrary Prolog atoms selected at random as being of various categories. The fact that speakers and hearers have equal access as witness to events models the assumption that original speakers and hearers had equivalent cognitive architectures and access to the same range of concepts. Similarly, the fact that speakers and hearers have access to the same inventory of phonemes creatively arranged into words by they themselves models the assumption that original speakers and hearers had comparable vocal tracts and auditory systems.

## B. Algorithms

1) *The basic architecture:* of the system involves iterating through the following process based on the input parameters.<sup>9</sup>

### 1) Initialize:

clear memory, etc.

generate enough phonemes <sup>$\pi$</sup> , entities <sup>$\pi$</sup>  and relations <sup>$\pi$</sup>

### 2) If out of Epochs <sup>$\pi$</sup> , show statistics & quit.

<sup>9</sup>A superscript  $\pi$  is supplied for each quantity that is parameterized.

- 3) If at the limit of attention span $^{\pi}$  (the end of an epoch) forget any symbol/meaning pair from the last epoch that occurred no more often than the forgetting threshold $^{\pi}$
- 4) Run an epoch.
  - a) Some arbitrary $^{\pi}$  relation obtains with appropriate arguments as an event
  - b) A speaker comments on that event, note taken
  - c) A hearer observes the event and interprets the utterance, note taken
  - d) Omniscients observe the degree of common understanding
  - e) If according to omniscients, the minimum<sup>10</sup> required level of understanding $^{\pi}$  is not reached, then go to 4c
  - f) If it's a situation falling under the probability that success matters $^{\pi}$ , and if success is less than a threshold percentage $^{\pi}$ , then ignore the utterance  
else, note the symbol/meaning association (updating frequencies), and go to 4a
- 5) Goto 2

This algorithm involves iterations of speakers commenting on events, and hearers interpreting utterances in the context of a jointly witnessed event. These subroutines are outlined below. Both speakers and hearers have the capacity to innovate. Both speakers and hearers are influenced by the history of communication (modulo that which is forgotten because it happened so infrequently). That influence of history may be frequency based (depending on all prior utterances) or recency based (depending on the most recent utterance). Because over time speakers and hearers will interchange roles and be equally influenced by their own past interpretations as by associations put forward through their utterances, there is a communal knowledge base of past interpretations. Essentially, this is a hardwired assumption that there are only two communicators. It is important to point out that in our idealization there is only one knowledge base of past utterances, modeling the pair's dual roles as speakers on some occasions and hearers on others.<sup>11</sup> This is not a model of differential memories among the participants, nor is there

<sup>10</sup>Recall that we feel the most appropriate level for this parameter is 0.

<sup>11</sup>One could well argue that this idealization is at best of an individual communicating with itself. To the extent that it's a valid argument, one must also agree that communicating with oneself is a nontrivial ability to have emerged. Recall, for example, [31] discusses a hypothesis that the origin of human consciousness is in the integration of the hemispheres, the end of interhemisphere intraindividual discrete communication; also recall the arguments on the modularity hypothesis [32] and Chomsky's minimalism that the role of language is not so much interpersonal communication as interfacing to the rest of cognitive architecture.

a real model of information state beyond the immediate event. A richer extension of the system would include participant-indexed knowledge bases and information states, with updates, downdates, queries, disputes and acknowledgments. Such an extension would demonstrate the effects of agent self-organization on the emerging language, but given that previous simulations (e.g. [25]) have shown that multi-agent systems will eventually exhibit self-organization and language coherence we do not feel compelled to attempt to replicate these results. We do recognize though that an attempt must be made to model the effects of lexical self-propagation among larger groups of agents. In simulations with many agents there will frequently be times when a particular agent will not have a particular type-token mapping or will only be aware of an archaic mapping even if there is general coherence within the group, thus we have parameters which determine the level of use of novel or infrequent mappings using either a frequency or recency based metric.

2) *Initialization:* The most interesting phase of the initialization is in the handling of Zipfian distributions of possible events. This is managed as follows.

- 1) Given the number of relation names <sup>$\pi$</sup>  to generate as  $E$ , initialize  $D = E$ ; compute the sum of series  $S$ :
  - a) Initialize  $S = 0$
  - b) If  $D = 1$  then return  $S = S + \frac{E}{D}$
  - c)  $S = S + \frac{E}{D}$
  - d) Decrement  $D$
  - e) goto 1b
- 2) The probability of the highest ranked event name is  $\frac{E}{S}$
- 3) The probability of each event name for rank  $r \geq 1$  is  $\frac{E}{S \cdot r}$

Each of the resulting event names is asserted to the database in an additional table indicating, in decreasing order, the Zipfian likelihood of an event of that sort occurring, along with the sum of probabilities up to that rank. Thus, as the Zipfian probability column decreases, the aggregate column approaches the value 1. Then, when an arbitrary event obtains, it is chosen with respect to a random value such that the event selected is the first event name in the table with an aggregate probability exceeding the randomly generated number (hence, the highest ranking event name according to the Zipfian distribution).

If the Zipfian parameter is not set, then Event names are chosen randomly and assigned to the seven event types without additional constraints on their frequency of occurrence in reality.

3) *Speaker comment on events:*

- 1) The event is modeled as the list of atoms given by a relation name and its arguments.
- 2) The speaker individuates that event.

This is modeled by:

- a) This list of possible partitions of the event list is formed.
- b) A random partitioning is selected from that list.

(This is a list of lists of atoms).

For example, an event obtains:

[mgvgmns, tkhs, xipz]

Possible construals of the relation:

[[ [mgvgmns], [tkhs, xipz]],  
 [ [mgvgmns], [tkhs], [xipz]],  
 [ [mgvgmns, tkhs], [xipz]]]

What the speaker focuses on:

[ [mgvgmns, tkhs], [xipz]]

- 3) The speaker associates a symbol/phoneme sequence with the structured perceived meaning.

This is in partial relation to what has been uttered and construed before.

For each meaning segment in the speaker's individuated meaning:

- a) Identify all the phonemes associated with it in the past
- b) If that list is nonempty:
  - i) Choose a random element of it.<sup>12</sup>
  - ii) If some chance event happens, whose probability diminishes with the number of words uttered (and remembered) so far, then just invent a new phoneme sequence to associate with the meaning.<sup>13</sup>

<sup>12</sup>This implements frequency effects: it is not the most frequent pairing that is necessarily selected, but frequent pairings have higher probability of selection from memory than infrequent pairings given that a random selection is made from the remembered history of pairings.

<sup>13</sup>Thus, innovation is possible at any stage, whether the speaker is influenced by frequency <sup>$\pi$</sup>  or recency <sup>$\pi$</sup> .

Otherwise, the uttered phoneme sequence is the random choice among past events. Otherwise, that meaning hasn't been noted before, so make up a new phoneme pair to associate with the meaning unit.

4) *Hearer interpretation of an event:*

- 1) The interpreter hears an event and segments the stream into units.
- 2) Interpretation associates meanings with the segments in relation to the event that occurred and the history of past associations
- 3) The interpreter is either frequency <sup>$\pi$</sup>  or recency <sup>$\pi$</sup>  sensitive.<sup>14</sup>
  - a) If there are no more "words" in the segmentation, association is done.
  - b) Find all past associations of meanings with the next word.
    - i) If frequency sensitive, select a random element of this list as the current interpretation of the word.
    - ii) Otherwise, if recency sensitive, select the last interpretation of the word as the current.
    - iii) If no past associations exist for the word, then on the basis of the hearer's own individuation of the event, associate a meaning-segment with the speech segment.
  - c) Goto 3a

Notice that our model assumes that speakers have more opportunities for innovation than hearers, but that both have opportunity to innovate.

### C. Discussion

In this section we have characterized the primary data structures and included details of their implementation in Prolog. We have also provided the main algorithms operating over those data structures as a function of choice in input parameters. Although the algorithms are described here in an imperative format, their mapping onto standard recursive clauses in a Prolog implementation is trivial.

## IV. SAMPLE EXPERIMENTS

Here we provide examples of typical experiments enabled by the system.

<sup>14</sup>One of our motives for providing a facility to explore these two parameter settings comes from the input-output coordination principle hypothesized as an explanation of empirical findings in the coordination of descriptions in task based dialog [33].

### A. A Benchmark

- Parameters:
  - Size of phoneme space: 30
  - Size of entity pool: 100
  - Size of relation pool: 30
  - Distribution of relation types: random
  - Number of epochs: 50
  - Attention span: 20
  - Forgetting threshold: 1
  - Minimum required level of understanding: 0
  - Probability that success matters: 0
  - Level of necessary success: 0
  - Sensitivity to recency vs. frequency: f

It was assumed that 30 phonemes were pronounceable and discernible. Only 100 entities could be discriminated, and merely 30 relations were noticed. The justification for these discernibility parameters is in presuming that the first speakers had limited cognitive abilities for discriminating entities and relations across time. We are not committed to such an assumption. Fifty conversations happened, each with 20 utterances (one for each of 20 events), every phoneme sequence mapping to meanings that failed to recur during the 20 conversations or all recall of past conversations being forgotten. There was no task driven probability for successful communication to matter and no minimal level of successful communication necessary for an utterance to be recorded. When relying on memory, speakers and hearers attended to more than just the last mapping of phonemes to a meaning (recency), but also the whole history of remembered mappings (frequency).

- Performance:
  - Average understanding of last 10 utterances: 67%
  - Average understanding overall: 25%
  - Number of words uttered: 4,531
  - Number of word types: 264
  - Homonymy: 15.4%

– Synonymy: 57.6%

Overall at the end, only 25% of what was uttered was understood, although 67% of was the average level of understanding over the final ten utterances. There were 4,531 individual utterances involving 364 remembered word types. Most meanings could be expressed by distinct strings of phonemes but overwhelmingly fewer strings could be used to designate more than one meaning. Refer to Figure 3 for a graph of the output: the X-axis provides the number of utterances and the Y-axis graphs the average understanding over last 10 utterances using the parameter settings mentioned above.

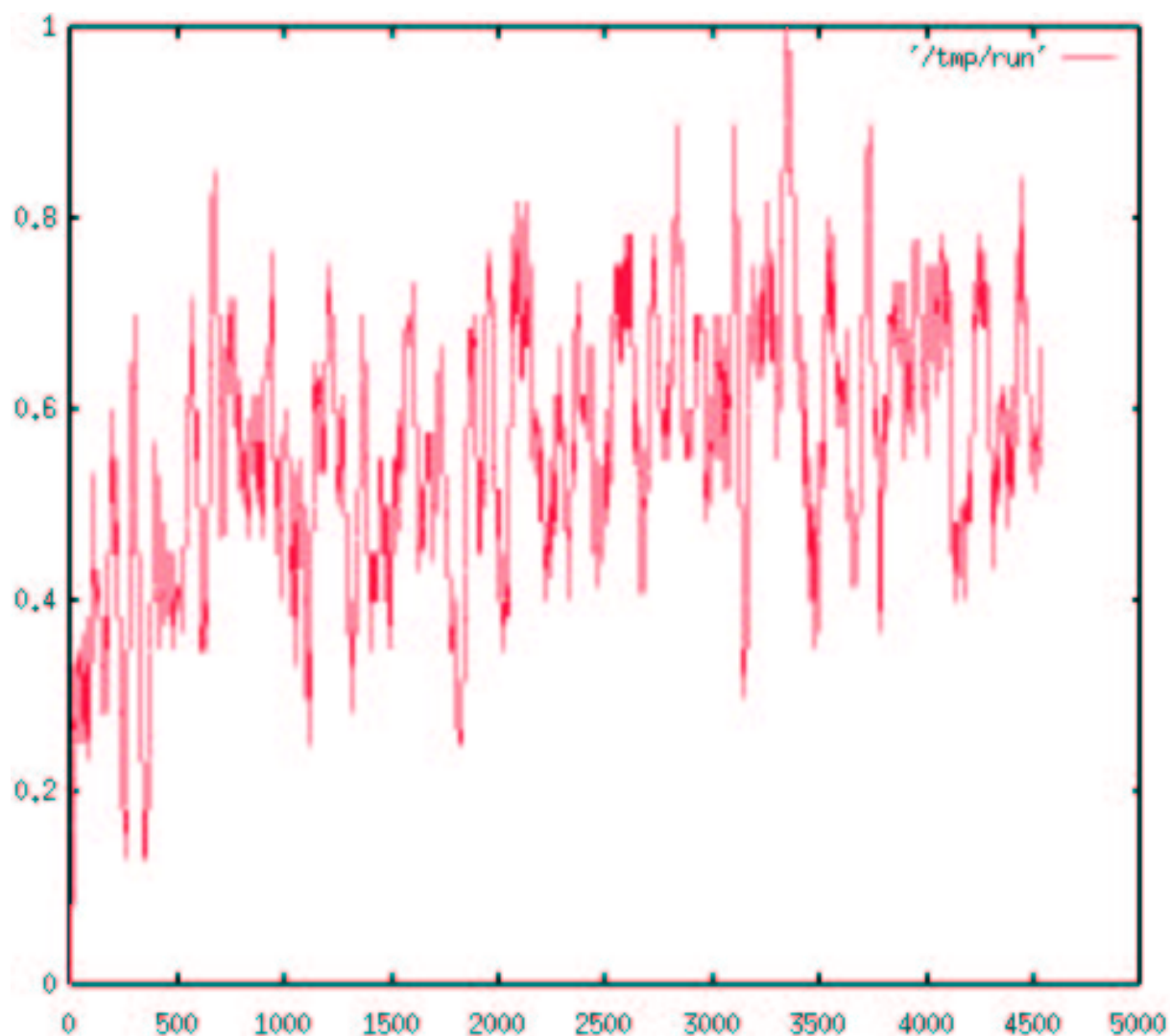


Fig. 3. Graphical output: percentage understanding vs. words uttered

Here we have set reasonable values to the various parameters. The result, even in the extreme pessimism about semantic transparency is fairly rapid achievement of more than 40% understanding of an utterance, 30% communicative success reached within the first 100 attempts at using representations for meanings — quite early on. A comparable graph of a run on identical inputs is provided in Figure 4. The performance as evaluated at the end of the run is as follows:

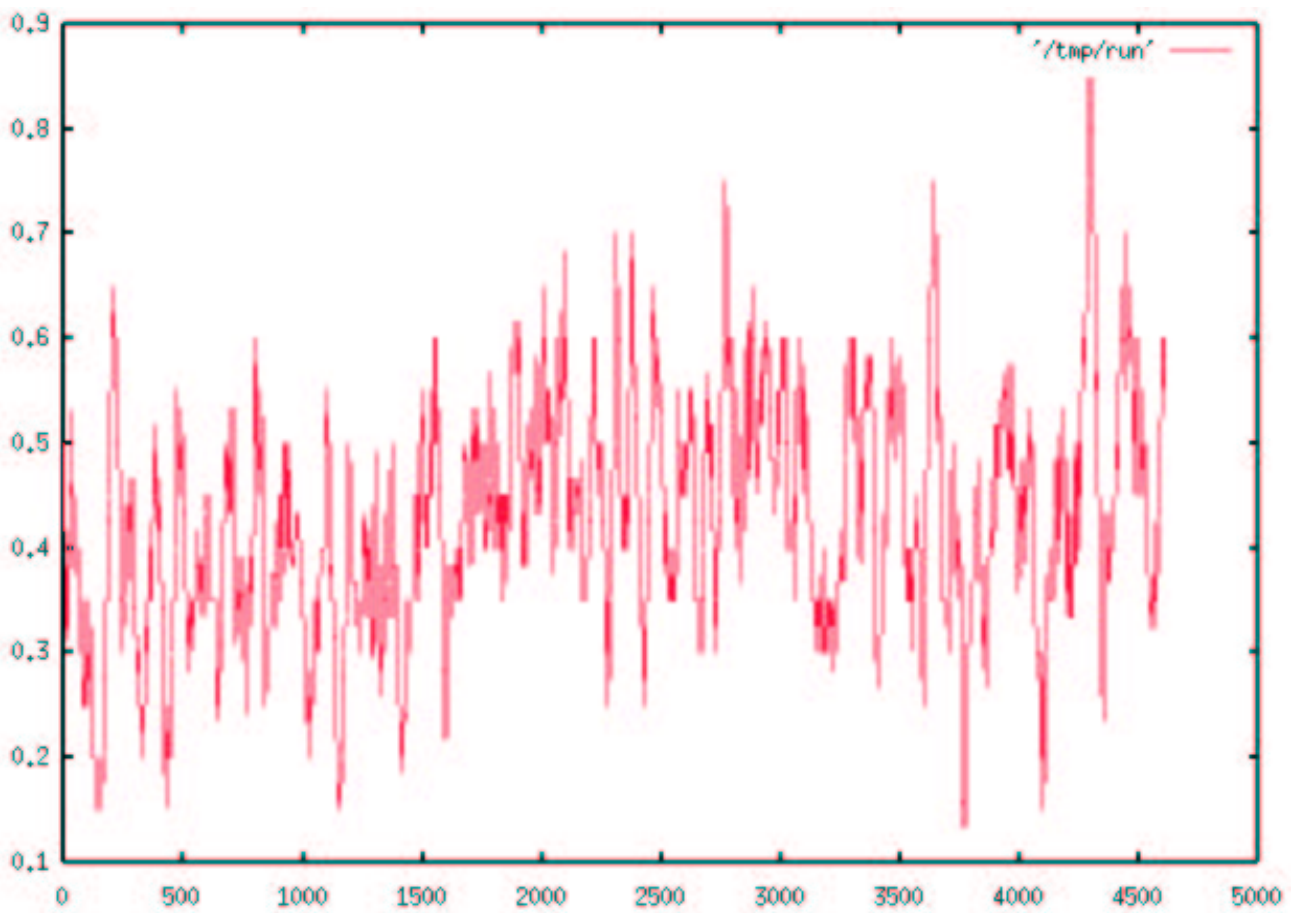
- Average understanding of last 10 utterances: 55%
- Average understanding overall: 42.8%
- Number of words uttered: 4,604
- Number of word types: 428
- Homonymy: 27.8%
- Synonymy: 49.3%

It is interesting as well to monitor additional features over the course of a simulation. Figure 5 shows the same run, but in a slightly different fashion — the X-axis is given by the total number of iterations ( $50 \text{ epochs/games} \times 20 \text{ conversations/attention span}$ ), and the Y-axis is the range between 0 and 1 of a number of the quantities we measure.<sup>15</sup> The time course of these measurements is useful to follow. Synonymy begins at quite a high value, and over time diminishes (here to about the same level as average overall understanding). Homonymy is more or less stable, at a level less than average understanding.

### *B. Varying Requisite Understanding Levels*

One parameter that we can set on the interval between 0 and 1 is the minimum required level of understanding for each utterance. Setting the level to 1, for example, presumes that learners have complete access to intended meaning from the start of language generation and learning. In general, we favor setting this parameter to 0. [15] examines a closely related pair of parameters (probability that success of any particular utterance matters, and the level of success necessary if it does matter) that involves levels of feedback on how well understood an utterance was. While as external observers we can monitor the degree of communicative success of our agents, unless that information is fed back to the agents they have no information about intended meanings. When, for example, the probability that successful communication matters to the

<sup>15</sup>In electronic versions of this paper, the lines which are labeled by arrows here are additionally color coded: red is recent understanding, green is average understanding overall, blue is homonymy and pink is synonymy.



September 5, 2002  
Fig. 4. Graphical output: percentage understanding vs. words uttered

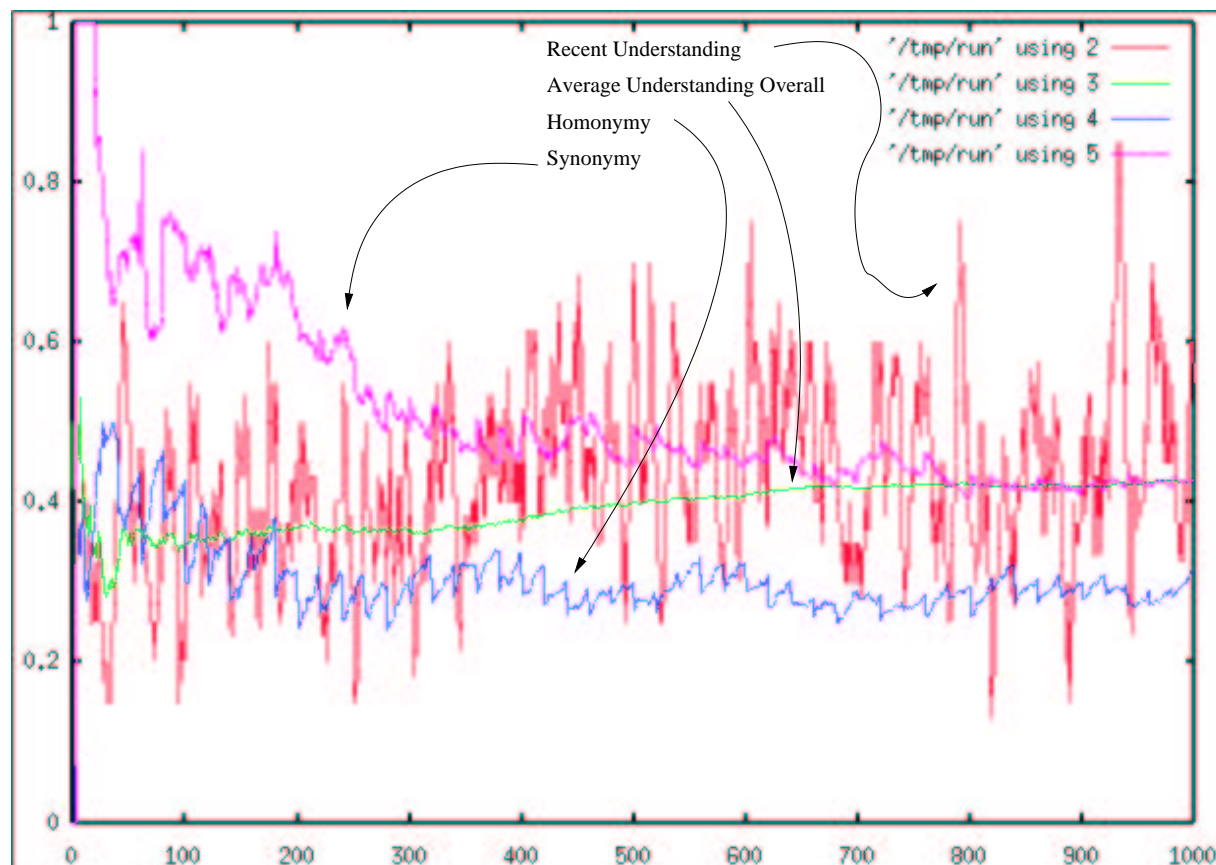


Fig. 5. Percentage understanding vs. homonymy vs. synonymy as a function of time

speakers, then not only do they lack access to each other's meanings, but further there is no task-oriented feedback which informs them that a breakdown in communication has occurred. Under that setting, we have a maximally pessimistic assumption about semantic transparency at the onset of language evolution. The feedback parameters are separate from the minimum level of understanding parameter in that, when the latter is set to 0, and the former pair (for example) stipulate that there is an 80 per cent chance of it mattering whether an utterance was at least 50 per cent understood, any individual communication involving 0 understanding may well fall into the chances that it didn't matter anyway. Equally, under the same feedback settings, if minimum understanding for each utterance is at the other extreme of 1, then the same communication will simply have to be understood, regardless of the feedback levels. Convergence on use of terms can emerge even if individuals assume only that they share meanings without actually sharing meanings, nor attending to evidence either way. Nonetheless, as omniscient observers,

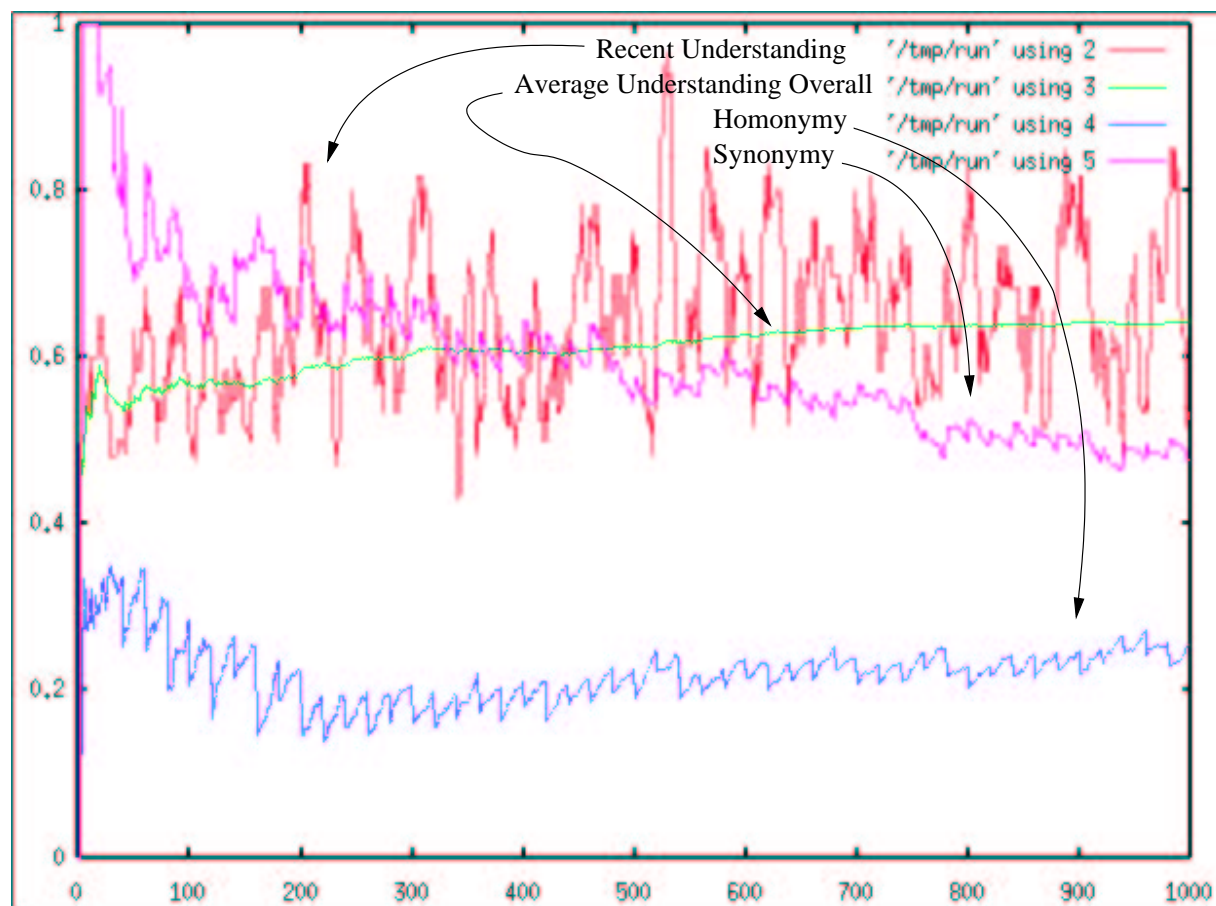


Fig. 6. Percentage understanding vs. homonymy vs. synonymy vs. time

we require the ability to assess whether meaning sharing has taken place and the degree to which it has. Success in communication is computed as an average over all communications and in the 10 most recent utterances.

Recall that the sample run of the system depicted in Figure 5 assumed complete pessimism about the level of understanding in each initial utterance: the parameter was set to 0. Figure 6 depicts a run with the value of all other parameters identical, but with the minimum understanding threshold set at 25%. This means that each utterance is attended to until it is at least 25% understood by the hearer. The level of understanding could exceed this minimum, of course, and in the results of the run it is evident that actual average understanding over time exceeded 60% (even without feedback), about 10% greater than the level of synonymy. We prefer holding this minimum understanding threshold at 0, in keeping with our preference for the maximally pessimistic assumptions on semantic transparency thus addressing misgivings about providing

agents with access to meanings [15]. Even with this pessimistic value, the run depicted in Figure 5 demonstrates overall successful communication of about 40%, that level of understanding achieved quite early on. For all values of the parameter we presume that it is essentially measured by omniscient observers, that the interlocutors do not know which parts of an utterance were successfully communicated when the threshold is less than 1.

### C. Discussion

This section has provided a sampling of the sorts of experiments that may be conducted using our workbench for simulating natural language evolution with respect to interesting parameters. The experiments were not exhaustive in their pursuit of parameter tuning. In fact, the evolution of an optimal language was not even sought, in that we hoped to demonstrate that even pessimistic assumptions can quickly lead to a viable language as a social construct. [14] report on interactions of parameters associated with articulable phoneme space. [15] consider memory and feedback issues in detail as well. [16] reports on research into the Zipfian distribution of event names, as well as a number of other interacting parameters.

## V. CONCLUSION

While at an advanced stage, our ongoing work includes enhancing the parameterization further to include representations of information states and dialog plans as well as memories of utterance-meaning mappings. We intend to extend the system to speech acts additional to assertion, including questioning and denial, along with accompanying information state updates. We intend these extensions to allow additional parameterization of the number of interlocutors represented as well as the subset of those included in any particular language game.

We have described a workbench that we use for exploring interactions of parameters in simulated natural language evolution from a social construct perspective. The system allows experimentation with some parameters that have not before been individuated. We have provided examples of the sorts of experiments made possible by the system.

## REFERENCES

- [1] James Hurford, "Biological evolution of the saussurean sign as a component of the language acquisition device," *Lingua*, vol. 77, pp. 187–222, 1989.

- [2] James Hurford, “The evolution of the critical period for language acquisition,” *Cognition*, vol. 40, pp. 159–201, 1991.
- [3] James Hurford, “Protothought had no logical names,” in *New Essays on the origin of Language*, Jurgen Trabant, Ed., pp. 119–132. Berlin: Morton de Gruyter, 2001.
- [4] James Hurford, “Random boolean nets and features of language,” *IEEE Transactions on Evolutionary Computation*, vol. 5, no. 2, pp. 111–116, 2001.
- [5] Angelo Cangelosi, “Evolution of communication and language using signals, symbols, and words,” *IEEE Transactions on Evolutionary Computation*, vol. 5, no. 2, pp. 93–101, 2001.
- [6] Simon Kirby, “Spontaneous evolution of linguistic structure—an iterated learning model of the emergence of regularity and irregularity,” *IEEE Transactions on Evolutionary Computation*, vol. 5, no. 2, pp. 102–110, 2001.
- [7] Chris Knight, Michael Studdert-Kennedy, and James Hurford, Eds., *The Evolutionary Emergence of Language*, Cambridge University Press, 2000.
- [8] James Hurford, Michael Studdert-Kennedy, and Chris Knight, Eds., *Approaches to the Evolution of Language*, Cambridge University Press, 1998.
- [9] Angelo Cangelosi and Domenico Parisi, Eds., *Simulating the Evolution of Language*, London: Springer, 2002.
- [10] Alison Wray, Ed., *The Transition to Language*, Oxford University Press, 2002.
- [11] Ágoston Endre Eigen, Robert Hinterding, and Zbigniew Michalewicz, “Parameter control in evolutionary algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 3, pp. 124–141, 2001.
- [12] Emma Hart and Peter Ross, “Gavel—a new tool for genetic algorithm visualization,” *IEEE Transactions on Evolutionary Computation*, vol. 5, no. 4, pp. 335–348, 2001.
- [13] David McFadzean, Deron Stewart, and Leigh Tesfatsion, “A computational laboratory for evolutionary trade networks,” *IEEE Transactions on Evolutionary Computation*, vol. 5, no. 5, pp. 546–560, 2001.
- [14] Carl Vogel and Justin Woods, “Simulation of evolving linguistic communication among fallible communicators,” in *Proceedings of the Fourth International Conference on the Evolution of Language*, James Hurford and Tecumseh Fitch, Eds. Harvard University, Cambridge, MA, 2002, p. 116.
- [15] Carl Vogel and Justin Woods, “Fallible communicators evolve successful communication,” Tech. Rep. TCD-CS-2002-31, Computational Linguistics Laboratory, Trinity College, University of Dublin, 2002.
- [16] Justin Woods, “Declaratives, interrogatives, semantic space and the emergence of communication,” M.S. thesis, Computational Linguistics Lab, Trinity College, University of Dublin, 2002, In Preparation; on Schedule for Completion September 2002.
- [17] Philip Johnson-Laird, “Mental models and deduction,” *Trends in Cognitive Sciences*, vol. 4, no. 10, pp. 434–42, 2001.
- [18] L. Steels and F. Kaplan, “Bootstrapping grounded word semantics,” in *Linguistic evolution through language acquisition: formal and computational models*, Ted Briscoe, Ed. Cambridge University Press, 1999.
- [19] Simon Kirby, “Syntax without natural selection: How compositionality emerges from vocabulary in a population of learners,” in *The Evolutionary Emergence of Language*, Chris Knight, Michael Studdert-Kennedy, and James Hurford, Eds., pp. 303–323. Cambridge University Press, 2000.
- [20] Steven Pinker, *Language Learnability and Language Development*, Harvard University Press, 1996.
- [21] Luc Steels, “Synthesizing the origins of language and meaning using co-evolution, self-organization and level formation,” in *Evolution of Human Language*, Jim Hurford, Ed. Edinburgh University Press, 1997.
- [22] Michael F. Schober and Herbert H. Clark, “Understanding by addressees and overhearers,” *Cognitive Psychology*, vol. 21, pp. 211–32, 1989.
- [23] E. M. Gold, “Language identification in the limit,” *Information and Control*, vol. 16, pp. 447–74, 1967.

- [24] Simon Garrod and Gwyneth Doherty, “Conversation, co-ordination and convention: an empirical investigation of how groups establish linguistic conventions,” *Cognition*, vol. 53, pp. 181–215, 1994.
- [25] L. Steels, F. Kaplan, A. McIntyre, and J. Van Looveren, “Crucial factors in the origins of word-meaning,” in *The Transition to Language*, Alison Wray, Ed., chapter 12, pp. 252–271. Oxford University Press, 2002.
- [26] James Hurford, “Social transmission favours linguistic generalisation,” in *The Evolutionary Emergence of Language*, Chris Knight, Michael Studdert-Kennedy, and James Hurford, Eds., pp. 324–352. Cambridge University Press, 2000.
- [27] Peter Ford Dominey, “Requirements on conceptual representation for the evolution of language,” in *Proceedings of the Fourth International Conference on the Evolution of Language*, James Hurford and Tecumseh Fitch, Eds. Harvard University, Cambridge, MA, 2002, p. 40.
- [28] Andrew Carstairs-McCarthy, *The Origins of Complex Language: An Inquiry into the Evolutionary Beginnings of Sentences, Syllables and Truth*, Oxford: Oxford University Press, 1998.
- [29] M. F. Schober, “Spatial perspective-taking in conversation,” *Cognition*, vol. 47, pp. 1–24, 1993.
- [30] Whitney Tabor, “The gradualness of syntactic change: A corpus proximity model,” 1993.
- [31] Julian Jaynes, *The Origins of Consciousness in the Breakdown of the Bicameral Mind*, Princeton, 1976.
- [32] Jerry A. Fodor, *The Modularity of Mind*, Cambridge: MIT Press, 1983.
- [33] Simon C. Garrod and Anthony Anderson, “Saying what you mean in dialogue: A study in conceptual and semantic co-ordination,” *Cognition*, vol. 27, pp. 181–218, 1987.