

# Expression/Induction Models of Language Evolution: Dimensions and Issues

James R Hurford  
Language Evolution and Computation  
Research Unit, Linguistics Department,  
University of Edinburgh

## 1.1 Introduction

Evolutionary modelling is moving into the challenging field of the evolution of syntactic systems. In this chapter<sup>1</sup>, five recent models will be compared. The following abbreviations will be used in referring to them.

Batali (1998)	JB1
Batali (this volume)	JB2
Hurford (in press)	JH
Kirby (in press)	SK1
Kirby (this volume)	SK2

Other related work will be mentioned where relevant<sup>2 3</sup>. The goals of the comparison will be to highlight shared and different assumptions and consequent shared and different outcomes.

The models of the evolution of syntax that have been constructed so far fall short of the kind of syntactic complexity found in real languages.

<sup>1</sup> This general kind of work was significantly stimulated by a workshop in computational evolutionary syntax sponsored by the Collegium Budapest Institute for Advanced Study in 1997. The particular work of this paper was substantially helped by a UK ESRC research grant, No. R000 237551. I thank Simon Kirby, John Batali, Ted Briscoe and Mike Oliphant for helpful comments, but I take sole responsibility for what is said here.

<sup>2</sup> The discussion of JB2 here is based largely based on a slightly earlier version than that published in this volume. Nothing significant turns on this.

<sup>3</sup> Steels (1998) outlines a model very similar in spirit to those compared here, but gives no details of any language-like system that is its outcome. For this reason, and for lack of space, it is discussed in less detail here.

In this work, idealization and simplification are immediately obvious. So far, the emergent language systems are, by the standards of extant languages, very simple. The models surveyed here all claim to present examples of the evolution from situations with no language to established syntactic systems. The evolved systems are admittedly simple, but this can be seen as a strength, rather than a weakness of these models, which abstract away from peripheral and incidental features of language, to focus on core properties such as compositionality, recursion and word order. As human syntactic ability has for long been held (by linguists) to be at the core of the innate language faculty, any claim to have simulated the evolution of some syntax needs to be evaluated with care. Questions that arise include:

- In what sense, and to what degree, do the evolved systems actually exhibit true syntax? This requires a theory-neutral definition of the term ‘syntax’.
- If some syntax is present in the evolved systems, to what extent is this syntax truly emergent, that is, neither simply programmed in nor an obvious consequence of the definitions of the central mechanisms (production and learning) or of the predefined semantic structures?
- In what ways do the evolved systems resemble natural languages?

After this introductory section, successive subsections will address these, and related, questions.

### **1.1.1 *Characteristics of Expression/Induction models***

‘Expression/Induction’, henceforth E/I, is a natural mnemonic for a class of computational models of language. In such E/I models, a language is treated as a dynamic system in which information is constantly recycled, over time, between two sorts of phase in the language’s life. In such a model, a language persists historically through successive instantiations in two quite different media: (1) mental grammars of individuals, and (2) public behaviour in the form of utterances (possibly affected by noise) paired with manifestations of their meanings (also possibly incomplete). In the history of a language, grammars in the heads of individuals do not give rise directly to grammars in the heads of other individuals; rather, grammars are the basis for an individual’s performance, and it is this overt behaviour from which other individuals induce their own mentally represented grammars.

There is nothing new in this view of language constantly spiralling between induced mental representations of its system (Chomskyan I-Language), and expressions of the system in behaviour (Chomskyan E-Language); it is essentially the picture presented by Andersen (1973), and assumed in many generative historical linguistic studies (e.g. Lightfoot, 1999). The term ‘E/I’ is deliberately reminiscent of the E-language/I-language distinction. However, the class of models I shall discuss under the rubric of ‘E/I models’ have certain further common features, listed in outline below.

**Computational implementation:** These models are fully implemented in computer simulations. They thus benefit from the clarity and rigour which computer implementation forces, while incurring the high degree of idealization and simplification typical of computer simulations. Obviously, the authors of these models, while admitting to the idealization and simplification, feel that the compensations of clarity and rigour yield some worthwhile conclusions.

**Populations of agents:** In these simulations, there are populations of individuals, each of whom is endowed with two essential capacities, given in the next two paragraphs below. During the course of a simulation, these agents are variously and alternately designated as speakers/teachers and hearers/learners. In a typical setup, every simulated individual has a chance of talking or listening to every other at some stage. In most models, the population changes regularly, with some individuals being removed (‘dying’) and new ones being introduced (‘being born’).

**Expression/invention capacity:** This is the capacity to produce an utterance, on being prompted with a given meaning. The utterance produced may be defined by a grammar already possessed by the individual, or be entirely generated by a process of ‘invention’ by random selection from the set of possible utterances, or else be formed partly by existing rules and partly by random invention. Where the individual’s grammar defines several possible utterances corresponding to a meaning, the individual’s production capacity may be biased toward one of these utterances, contributing to a ‘bottleneck’ effect (see below).

**Grammar induction capacity:** This is the capacity to acquire, from a finite set of examples, an internal representation of a (possibly infinite) language system. A language system is a mapping between meanings and forms, equally amenable for use in both production and perception. The set of possible internalized grammars is constrained by the individual's acquisition algorithm. Furthermore, the individual's acquisition algorithm may bias it statistically toward one type of grammar in preference to a grammar of another type. Where the individual's acquisition device is a neural net, one may still speak of an internalized grammar, envisaged as the mapping between meanings and utterances reflected in the input/output behaviour of the trained net.

**Starting from no language:** These models focus on the question of how incipient languages could possibly emerge from situations in which language is absent. At the start of a simulation, the members of the initial population have no internalized representations of a particular language. The simulations nevertheless usually end with populations whose members all have substantial (near-)identical mental representations of some language, and all produce utterances conforming to a standard applying to the whole community. These models are thus not primarily models of historical language change in relatively mature and complex languages, although the methodology of these simulations, extended and refined, would be very suitable for models in historical linguistics. (Examples of such applications to historical language change, from quite contrasting theoretical backgrounds, are Hare and Elman (1995) and Niyogi and Berwick (1997)).

**No biological evolution:** In these models, there are no differences between individuals at the point when they are introduced into the population. They all have identical capacities for responding to their environment, either in the production of utterances or in the acquisition of an internal language system triggered by exposure to the utterances of others. Thus these are models of the cultural evolution of learned signalling systems, a quite special case of historical language change, as noted above. These models are not models of the rise of innate signalling systems.

**No effect of communication:** These models are clearly inspired by situations in which humans communicate meanings to each other. It

is in fact possible in these models to measure the degree to which the emergent systems allow successful communication between agents (JB2, in particular, emphasizes this). And the states on which the models converge would typically allow efficient communication. But raw communicative success is not a driving force in these models. That is, there is no instance in which a simulated speaker attempts to communicate a meaning, using a particular form, and then, noting how successful the attempt is, modifies its basis for future behaviour accordingly. The basic driving force is the learning of behaviour patterns by observation of the behaviour of others. The fact that the behaviour concerned can be interpreted as communicative, and that communication may happen to be beneficial to a group, is not what makes these models work. These are models of the process by which patterns of behaviour (which, quite incidentally, happen to be communicative) emerge among agents who acquire mental representations determining their own future behaviour as a result of observing the behaviour of others. The undoubtedly interesting and significant fact that such patterns of behaviour may convey selective advantage on individuals or populations that possess them is no part of these models.

**Lack of noise** Unrealistically, all the models surveyed here are noise-free. That is, every utterance produced by a speaker is assumed to be perfectly observed by a learner. Similarly, learners are assumed to have perfect access to the meanings expressed by their ‘teachers’. Thus these models do not deal with an obvious and potent source of language change. Nevertheless, leaving noise out of the equation, at least temporarily, serves a useful purpose, in that it allows us to see the evolutionary effects of other factors, such as bottlenecks (see below), all the more clearly. Perfect access to primary linguistic data is a basic assumption of classic work in language learnability theory and related theory of language change (e.g. Clark & Roberts (1993); Gibson & Wexler (1994); Niyogi & Berwick (1997)) . It is not a problematic assumption, because it is clear that it could be relaxed to partial access all of the time, or perfect access some of the time (or both), so long as such access is sufficient.

**Pre-defined meanings:** The extant models all take as given some set of pre-defined meaning representations. Such representations can be seen as thoughts, ideas or concepts, which the pre-linguistic agents can

entertain, but not express. In the course of a given simulation, the set of available meanings does not change, although their expressibility in utterances changes, typically from 0% to 100% . The pre-defined meanings are always structured and somewhat complex. The contribution of such semantic structure to the emergent language systems will be discussed in detail later.

**Pre-defined ‘phonetic’ alphabets:** The extant models all assume some unchanging finite vocabulary of atomic symbols from which utterances are constructed, by concatenation. The size of this vocabulary relative to the meaning space is an important factor.

**Emergence:** All such models aim to show how certain features of language emerge from the conditions set up. A major goal is to demonstrate the emergence of features which are not obviously built in to the simulations. This presupposes that the essential dynamic of an E/I model itself produces certain kinds of language structure as a highly likely outcome. The interaction of assumptions produces non-obvious outcomes explored by simulation. Actual models differ in the extent to which various structural properties of the resulting language system can be said to be built in to the definitions of the crucial processes in the simulation cycle.

**‘Bottlenecks’:** An individual’s acquired grammar may be recursive, and define an infinite set of meaning-form pairs, or, if not recursive, it may nevertheless define a very large set of meaning-form pairs. The set of example utterances which form the basis for the acquisition of an internal representation of language in an individual is necessarily finite (as is life). A bottleneck exists in an E/I model when the meaning-form pairs defined by an individual’s grammar are not presented in full as data to learners. A subset of examples from the infinite (or very large) range of the internalized grammars of one set of speakers is fed through a finite bottleneck to constitute the acquisition data of a set of learners. The simulation may, by design, prompt individual speakers with only a (random) subset of the available meanings, so that the data given to an acquirer lacks examples of the expression of some meanings. I will label this a **‘semantic bottleneck’**. With a semantic bottleneck, learners only observe expressions for a fraction of all possible meanings. Even where all individuals are systematically prompted to express all available

meanings (possible only where the set of meanings is finite), the individual speakers' production mechanisms may be designed to produce only a subset of the possible utterances for those meanings as defined by their grammars. We will label this a '**production bottleneck**'. Note that it would in fact be unrealistic **not** to implement a production bottleneck. Communication in real societies involves singular speech events, in which a speaker finds a single way of expressing a particular meaning. There is no natural communicative situation in which a speaker rehearses **all** her forms for a given meaning. It is the kind of metalinguistic exercise that might be part of fireside word games, or perhaps be used in a second language classroom, but nowhere else.

### 1.1.2 Simple examples: *Evolution of vocabulary*

To outline the basic shape of an E/I model, and to demonstrate the potential effects of bottlenecks in simple cases, we will start with the case of the evolution of a simple vocabulary. A number of earlier studies (Oliphant, 1997; Steels, 1996a, 1996b, 1996c, 1997; Vogt, 1998) model the emergence of simple vocabularies. Some of these vocabulary models technically satisfy the criteria listed above for E/I bottleneck models, and in doing so, illustrate some basic effects of the dynamics of these models. It is characteristic of models of vocabulary evolution that they assume a finite set of unrelated, atomic meanings. The lack of structured relationships between and inside vocabulary items ensures that each meaning-form pair must be acquired individually, and the whole lexicon is memorized as an unstructured list, over which no generalizations are possible. (The following informal examples are composed for this paper and representative of the literature, though not drawn wholly from any single publication.)

#### *Learned vocabulary transmission without bottlenecks*

Take a population of, say  $P$  individuals, each with access to a finite set of concepts, say  $C$  in number, and none, as yet, with any known means of expressing these concepts. Let each individual now try to express every concept to every other individual, uttering a syllable drawn from a large set. At first, no individual has any acquired means of expressing any concept (no mental lexicon), and so each resorts to invention by random selection from the large set of syllables. Let us say that the set of syllables is so large that, over the whole population, there are

likely to be few chance repetitions of the same syllable for the same meaning. The typical experience of an individual hearer/learner will be to hear  $p$  ( $p < P$ ) different syllables for each of the  $C$  concepts, and he will thus acquire a large lexicon consisting of  $p \times C$  meaning-form pairs. Across the whole population, there will be a great variety of such lexicons, overlapping with each other to some small degree. Now ‘kill off’ a fraction of the population; this reduces the linguistic diversity somewhat, but not much. Introduce a corresponding number of new individuals as learners, to whom all the surviving individuals will express **all** the available concepts, and, moreover, using **all** the syllables they have acquired for each of those meanings. Thus, the newly introduced learners are in fact exposed to the **whole** language, which they will acquire *in toto*, and in due course pass on *in toto* to the next generation. After the random inventions of the initial generation, there will be no further change, either in the internalized lexicons of successive members of this hypothetical community, or in its public language, for the rest of its history.

This is a situation with no bottleneck. A community which transmits its language without bottlenecks preserves an aboriginal set of meaning-form pairs down through the ages with comparable fidelity (but not by the same mechanism) as a community with an innate signalling system. After the invention of the original meaning-form pairs, there is no evolution in such a system.

*Vocabulary transmission with only a production bottleneck*

Let us now modify the scenario, and introduce a production bottleneck, but not, at this stage, a semantic bottleneck. A production bottleneck exists when a speaker has learned several forms for a given meaning, and selects among them when prompted with that meaning, with the result that some acquired forms are never uttered for this meaning by this speaker. We assume that all agents in a simulation apply the same selection method in implementing a production bottleneck. Some possible production bottleneck selection methods are:

- for a given meaning, use the form that was most frequently used for that meaning in your learning experience,
- for a given meaning, use the form that was first acquired for that meaning,
- use the shortest form in your vocabulary for the given meaning,
- use the form that is cheapest, according to some well defined metric,



- pick a random form from your vocabulary for the given meaning.

The last (random) method here has a special status as it assumes a uniform distribution; it is the weakest assumption about sampling, and all the others will set up some kind of positive feedback between learning and production, so that an explanation of emergent properties is no longer totally in terms of the learning algorithm. A selection method can also be probabilistic, not necessarily, but still possibly, eliminating the use of some dispreferred form. With a non-random production bottleneck implemented, each speaker is consistent over time in his method of choice of expressions for particular meanings.

It is apparent that, whatever selection method is used, the effect will be, over the course of many simulation cycles, to narrow down the set of forms used for a given meaning, until eventually there is only one form for a given meaning used across the whole community. Even if the selection of forms by agents is genuinely random, there will still occasionally be chance instances of some learner not acquiring some particular form-meaning pair, because this particular form happened not to have been used by any of the speakers from whom he learned; and such a form will be rarer in the E-Language of the next generation. If the population is a genuine single population, rather than several subpopulations completely isolated from contact with each other, then over time the number of forms for a given meaning will approach, and finally reach, 1. Even with a large population, if the production behaviour of an agent at one time may in principle historically affect, through the constant cycle of expression/induction/expression/induction, the learning of any agent at a much later time, then the population is guaranteed to converge, sooner or later, onto a common vocabulary, with just one form for each meaning. That is, the E/I model, with a production bottleneck, but no semantic bottleneck, leads inevitably to the elimination of synonyms in a language system. There is no corresponding tendency or mechanism for the elimination of homonyms. This is because, with no semantic bottleneck, the system is driven by meanings; it is a requirement of this condition that all the meanings be expressed to each learner each generation.

*Vocabulary transmission with only a semantic bottleneck*

Although the absence of a production bottleneck is unnatural, we will briefly consider the converse situation, in which an E/I model contains a semantic bottleneck, but no production bottleneck. In such a situation,

the meanings that are expressed by speakers each generation are a subset of the available meanings. In a given generation, some meanings will be picked for expression which were not picked in the previous generation, or vice-versa. Some meanings may be passed over for a period long enough to lead to expressions for them being lost from the grammars of all speakers (although all speakers are still capable of conceiving of the meanings). In this event, the first speaker called upon to express a particular meaning for which he has acquired no paired form will randomly invent a new form, which will then enter the language and be transmitted to subsequent generations. If several speakers in the same generation invent new forms for a given meaning, they will almost certainly invent different forms, and a case of synonymy is created. If the semantic bottleneck is especially fierce, with frequent omission of meanings, such re-invention will be constant, leading to an unstable language system with multiple synonymy evident at all stages.

It can be seen that, at the level of vocabulary, the two kinds of bottleneck are in tension with each other with regard to the phenomenon of synonymy. A semantic bottleneck tends to increase synonymy, by frequently triggering (re-)invention of forms; a production bottleneck tends to reduce synonymy, by guided selection of forms. We shall later see some echoes of these effects in the evolution of syntax, but with a crucial difference, in that the availability of general syntactic rules can preempt an appeal to random (re-)invention of forms.

#### *Frequency effects*

A model may have a near no-semantic-bottleneck condition, even without an explicit stipulation in the code that every meaning be expressed to every learner. This can happen if the selection of meanings to be expressed is random, but the number of available meanings is small in relation to the number of ‘speech-events’ typically experienced by each learner. In such a case, the probability of a meaning being omitted at some stage in the cycle is small, but real, and one can expect some occasional instability to result, with some form-meaning pairs disappearing from the system and being replaced by newly invented ones.

If we manipulate the frequency with which meanings are chosen to be expressed, so that some meanings are expressed so seldom that whole generations can pass without these meanings being expressed, there will be greater instability in the form-meaning pairings for these less used meanings, with frequent re-invention of new forms. On the other hand,

meanings that are expressed with high frequency (i.e. at least once to each learner every generation) will preserve their ancestral forms without change. Thus, in general, conservative form-meaning pairings are correlated with high frequency of use.

The principles underlying the evolution of vocabularies in situations with cyclic production and learning are now fairly clear. The phenomena encountered in this section on the emergence of simple vocabularies foreshadow, *mutatis mutandis*, phenomena found in the emergence of syntactic systems in E/I models. The introduction of syntactic capability brings new, and more interesting, features to language systems transmitted by E/I dynamics.

## 1.2 Modelling the emergence of syntax

### 1.2.1 Does true syntax emerge in E/I models?

A syntactic system is here defined to be any system in which strings of concatenated symbols are paired with somewhat complex meanings, by any systematic means other than exhaustive listing of the one-to-one meaning-form pairs. The mere fact that the forms are strings and that the meanings are complex is not sufficient to define a system as ‘syntactic’. With a truly syntactic system, what is important is that the **structure** of the expressions is part of what conveys meaning. Clearly, finite sets of meaning-form pairings involving symbol strings and complex meanings **could** evolve, and persist in a community, in exactly the same way as was outlined above for vocabularies with atomic meanings and forms. It will be seen, however, that, even with agents who are able (and sometimes even prefer), to rote-memorize individual correspondences between strings and complex meanings, there are pressures inherent in E/I bottleneck models which lead to the emergence of syntactic systems in the sense defined.

With symbolic (as opposed to connectionist) models (i.e. SK1, JH, SK2, JB2) there is a relatively direct method of diagnosing whether a system emerging from an E/I simulation is truly syntactic. This direct method is by inspection of the internalized grammars of the simulated agents, given an understanding of how these grammars are used to produce forms for target meanings. As a general working principle, if the number of separate statements in a grammar is less than the total number of meaning-form correspondences defined by the grammar, then some degree of syntactic generalization is present in the system.

This principle relies on an intuitive grasp of what should be counted as a separate statement in a grammar. In SK1, SK2 and JH, separately stored rules are clearly identifiable; in JB2, each stored exemplar is a separate statement of the grammar. The following table shows that, by this criterion, all the models surveyed have evolved syntactic means of expressing their meanings.

Model	Language size		Grammar size		
	Actual	Principled	Lexical	Phrasal	Total
SK1	100	100	10	1	11
SK2a	100	100	10	1	11
SK2b	65,000	infinite	15	2	17
JHa	4,137	infinite	16	3	19
JB2	23 trillion	infinite	35	305	340

Comparing language size to grammar size in emergent E/I systems. **Key:** SK2a = 1st experiment in SK2; SK2b = 2nd experiment in SK2; JHa = 1st experiment in JH; Language size (Actual) = Number of meaning-form correspondences defined within practical size limit imposed on simulation; Language size (Principled) = Number of meaning-form correspondences defined in principle; Grammar size (Lexical) = Number of most atomic meaning-form statements in grammar; Grammar size (Phrasal) = Number of rules or exemplars in grammar; Grammar size (Total) = Total number of statements in grammar. Figures for JB2 are approximate and from an early version of his paper.

In the case of the relatively orthodox phrase structure grammars of SK1 and SK2, if an agent has acquired any rule which, by including one or more variables, generalizes over more than one meaning-form pair, then that agent has at least some syntax. Agents may possess syntax to varying degrees, as shown by the intermediate phases of many of the simulation runs surveyed here, in which agents have some ‘partially syntactic rules’. A partially syntactic rule contains one or more variables which generalize over parts of a sentence, but also contains constant elements of both meaning and form, which are thus correlated only by the fact of their co-occurring in this rule, as in a lexical entry. An example of such a partially syntactic rule, from SK2, is reproduced below:

$$\begin{aligned}
 S/\text{loves}(\text{john},x) &\rightarrow \text{johnloves } N/x \\
 N/\text{mary} &\rightarrow \text{mary} \\
 N/\text{jane} &\rightarrow \text{jane}
 \end{aligned}$$

In the first of these three rules, the variables  $N$  and  $x$  generalize over the forms and meanings, respectively, seen in the other two (lexical) rules. At least two such other rules are necessary in the grammar for the generalization in the first rule to have any generalizing effect. Given such a combination, one may legitimately claim that the system containing them is at least incipiently syntactic in nature<sup>4</sup>.

Note that the presence, in an agent's internal grammar, of some apparently rote-learned correspondences between complex meanings and strings of symbols, with no generalizing variables, does not imply that some of those correspondences are not also the subject of generalization by rules also present in the grammar. The internal grammars of agents may be redundant, specifying the same facts about the emergent language system in more than one way. This is true, for example, of the system emerging from the third experiment described in JH, in which the capacity of learners to internalize generalizations was deliberately somewhat (but not totally) impaired.

### 1.2.2 Phases in the emergence of syntax

SK1 and JB2 both present graphs of typical runs which reveal three distinct phases in the emergence, from nothing, of syntactic systems. The first phase could be termed a 'holistic' phase. It is described by the modellers as follows:

The grammars at this stage are basically vocabulary lists, with each complex meaning being expressed as an arbitrary unanalysed string of symbols. . . . there is no consistent way in which the meanings are related to the strings.

*SK1*

Each utterance is analysed as a token, and such tokens are worthless for expressing anything but the exact meaning they contain.

*JB2*

The second phase could be termed 'uncoordinated transitional'. It is described as follows:

<sup>4</sup> The reservation implicit here arises from the fact that, in the example given, only two sentences are generated by three statements; adding further lexical items to fill the  $N$  slot does not help, as there will still always be one more statement in the grammar than there are sentences generated.

The number of meanings covered increases dramatically, as does the size of the grammar. . . the grammars at this stage are far more complex and byzantine than the earlier ones.

*SK1*

. . . the agents begin to acquire and use complex phrase exemplars. This is followed by rapid accumulation of exemplars . . .

*JB2*

The third phase could be termed ‘stable economical’. It is described as follows:

The transition [to stage 3] is marked by a sudden increase in the number of meanings that can be produced to the maximum value and a drop in the size of grammars

*SK1*

The average number of token exemplars, which by round 8000 almost all contain a single formula as their meaning, decreases from its peak of 220 to 35 by round 25000.<sup>5</sup>

*JB2*

The similarities between the typical runs of SK1 and JB2, despite the markedly different structures and assumptions of the models, are suggestive. Further work needs to be done to see whether these three phases are in any sense necessary to the emergence of syntax. SK2 does not report such distinct phases, which may be a consequence of using the minimal 2-agent (1 ‘teacher’, 1 learner) population, thus eliminating the factor of coordination of grammars within a population. The runs in JH converge on syntactic systems so rapidly that it is not possible to discern any middle or transitional phase. The final version of JB2 (in this volume) actually distinguishes four phases, with the third ‘stable economical’ phase above split into two. See Batali (this volume) for discussion.

### **1.2.3 Agents’ representations of syntax**

It is useful to mention that for a system to be called ‘syntactic’ it is neither necessary nor sufficient that the internalized grammars of agents

<sup>5</sup> This is quoted from an early version of Batali’s paper. The figures for the final model described in this volume are somewhat different, but nothing significant turns on this.

contain symbols that are interpreted as autonomous of both meaning and form. One of the central claims of generative grammar is that syntax is autonomous, that is, the terms used to characterize the syntactic system of a language are neither purely semantic nor purely phonetic, but are to some degree independent of both meaning and form. The thesis of the autonomy of syntax is an empirical claim, subject to falsification by analysis of actual languages. Thus, it cannot be a matter of definition that syntax is autonomous of form and meaning. We should define ‘syntax’ independently of the notion of autonomy. Having done so, we can then judge whether the systems evolved in E/I models do exhibit syntax. And, if they do, we can then ask whether, and in what sense, these evolved syntactic systems are autonomous of form and meaning.

It is of course possible, though not desirable, to write very complex phrase structure grammars, with a wealth of different syntactic categories, for very simple finite data, for which an intuitively better account would be provided by a list. If a simulation in an E/I model, through some quirk of its expression and induction algorithms, happened to culminate in a situation where agents had internalized complex grammars with such unjustified ‘syntactic’ structuring, we should not conclude that true syntax had emerged in this model, despite the grammars containing what look like syntactic category symbols.

The models of SK1 and SK2 contrast with those of JH and JB1 and JB2 in the degree to which learners postulate autonomous syntactic structure. The agents in SK1 and SK2 models induce context free phrase structure grammars with symbols representing autonomous syntactic categories. Apart from one specially designated symbol, *S*, for ‘sentence’, these syntactic category symbols are simply integers, generated as needed by the induction algorithm (and translated in the examples of SK2 as italic capital letters, *A*, *B*, *C*, ...). The agents thus have the facility to represent generalizations over classes of atomic forms (‘words’) and over classes of strings of atomic forms (‘phrases’), on the basis of their distribution in sentences, and of the systematic mapping of such classes to semantic terms. But the mere existence in an agent’s grammar of an apparently autonomous syntactic symbol, such as *S*, does not imply that the agent has acquired a syntactic system. The early grammars induced by the agents in SK1 and SK2 are essentially lexicons, in which the symbol *S* is present, but does no work. But the grammars which emerge later are clearly syntactic, by our criterion.

In the JH model, agents acquire grammars in which the rules have no

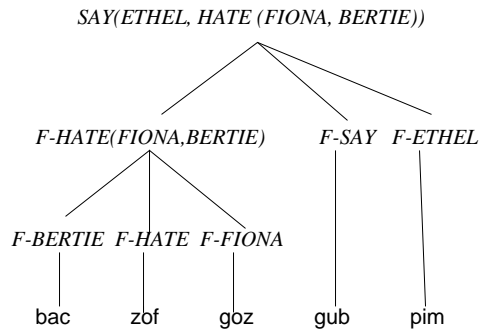
syntactic category symbols. There are dictionary statements, relating atomic meanings to atomic forms, that is translating predicates and individual constants into words, for example:

$$\begin{aligned} SING &\rightarrow \text{vag} \\ FIONA &\rightarrow \text{goz} \end{aligned}$$

Such lexical statements contain no variables. In addition, there are rules defining the order in which the components of a proposition are expressed, for example:

$$\{PRED, ARG1, ARG2\} \rightarrow \text{F-ARG2 F-PRED F-ARG1}$$

Here, 'F-' is an operator meaning 'form of' (as defined by other rules, including lexical, rules); all the other terms are variables; the left hand side of the rule depicts an unordered set of the identifiable parts of a proposition; and the right hand side of the rule states the linear order in which the defined forms occur. Such grammars contain no syntactic category symbols. But, by the definition of syntax followed here, the emergent systems are clearly syntactic, achieving a high degree of generalization over meaning-form correspondences. A tree diagram of the derivation in this grammar of a string expressing a complex meaning would look like this:

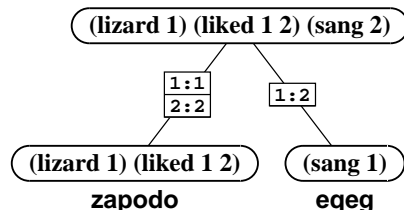


Such a tree structure contains no nodes labelled with specifically syntactic categories.

In JB2, as the table above shows, the emergent systems also achieve a high degree of generalization over meaning-form correspondences, and thus satisfy our criterion for having evolved some syntax. In this model



the representations induced by agents are not rules of any familiar kind, but ‘exemplars’. An exemplar, in JB2, is a more or less simple bit of tree structure whose terminal nodes are the syllable strings of the emergent language, and all of whose preterminal nodes are parts of semantic representations. An example is given below:



In the absence of autonomous syntactic categories, such tree structures resemble the previous one, derived in the JH model. The two structure types also, of course, resemble each other, and the tree structures derived in SK1 and SK2 in their progressive decomposition of a complex semantic representation, as one descends the tree, but this compositionality is to be expected of any truly syntactic system.

More important are the differences between SK1, SK2, JH, on the one hand, and JB2. In SK1, SK2 and JH, tree structures are not represented in an agent’s grammar, although they can be constructed by an analyst from the agent’s rules, whereas the JB2 exemplars are actually what the agents’ grammars consist of; in JB2, agents do not store rules.

The power of a grammar to define a large set of meaning-form correspondences in a few statements resides either in the grammar itself or outside the grammar, in the processing conventions or algorithms used to ‘read’ the grammar when producing forms for target meanings. This difference echoes the debate, in the early days of generative grammar, on the rival merits of **rules** and **analogies**.

Linguists have had their share in perpetuating the myth that linguistic behaviour is ‘habitual’ and that a fixed stock of ‘patterns’ is acquired through practice and used as a basis for ‘analogy’. These views could be maintained only as long as grammatical description was sufficiently vague and imprecise.

*Chomsky (1971/1965:154)*

*Analogy*, multiplied over and over, is the process by which a grammatical rule is formed.

*Bolinger, 1968:114*

The agents in JB2’s evolved systems, have acquired a fixed stock of

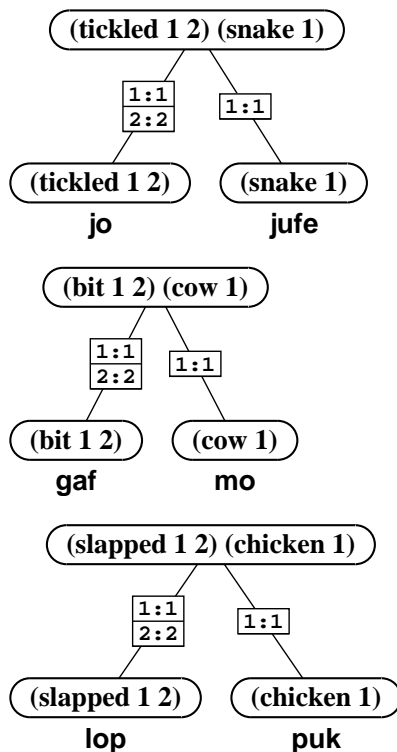
patterns (exemplars) and use them as a basis for analogy in producing new forms. They do not progress to the stage described by Bolinger in actually forming general grammatical rules. The term ‘exemplar’ itself implies that the agents’ stored representations are illustrative rather than generative. Variables are the explicit instruments of generalization. The exemplars of JB2 actually contain variables, the numbers 1 and 2, which are the arguments of predicates. However, the use which is made of these variables is not to characterize classes of expressions, but rather to define possible transformations of individual exemplars. The main generative power of the emergent systems in JB2 does not reside in the variables in the agents’ grammars. The JB2 model fits very closely with the ‘instance-based’ style of learning algorithm characterized by Langley as follows:

*... instance-based or case-based learning represents knowledge in terms of specific cases or experiences and relies on flexible matching methods to retrieve these cases and apply them to new situations. One common approach simply finds the stored case nearest (according to some distance metric) to the current situation, then uses it for classification or prediction. The typical case-based learning method simply stores training instances in memory; generalization occurs at retrieval time, with the power residing in the indexing scheme, the similarity metric used to identify relevant cases, and the method for adapting cases to new situations.*

*Langley (1996:21)*

JB2’s evolved agents end up with an average of about 340 exemplars, of which about 35 are lexical, or ‘tokens’ stating the most atomic meaning-form correspondences. This means that the remaining roughly 300 exemplars are complex or phrasal, and it is clear that many such stored phrasal exemplars are quite similar to each other. There is likely to be, for example, a whole set of separate exemplars along the lines of the three shown on the next page. In the classical parlance of generative grammar, a grammar which contained a set of such similar forms is ‘missing a generalization’. The obvious generalization could be captured by using variables over predicates. It is not necessarily a criticism of the JB2 model that its agents fail to capture a generalization. What is interesting, and more important, is that the population of agents has converged on a set of representations **over which a generalization is possible**. It is a notable general property of E/I models that they converge on systems over which generalization is possible, even where the agents themselves do not represent such generalizations internally. The

third experiment of JH also involves agents who represent their emergent language in a redundant, non-general way, but whose behaviour nevertheless has converged on a system over which strong generalizations are possible.



The intellectual background of the JB2 model is a view of language which is less Chomskyan than those of SK1, SK2 and JH, in its emphasis on exemplars, rather than generative rules. Here is not the place to discuss the empirical psycholinguistic issue of the degree to which humans store exemplars rather than rules. Certainly, the issue is not as cut-and-dried as many generativists perhaps believe.

... there have always been pockets within linguistics, sociolinguistics, and applied linguistics which have suggested that ready-made chunks of unanalysed language are as important as productive rules (Bolinger 1976; Coulmas 1979, 1981; VanLancker 1975; Widdowson 1984, 1990; Yorio 1980). Peters (1983) and more recently Nattinger and DeCarrico (1992) suggest that the role of ready-made chunks of language in L1 and L2 development may be underesti-

mated.

*Weinert, 1995:180*

The issue is that of the ‘formulaicity’ of language organization in the brain, and is closely related to the issue of holistic utterances in language evolution, discussed by Wray (1998, forthcoming). The issue of rules versus whole stored chunks also arises in computational parsing theory (e.g. Bod, 1998). The fact that computational models with such contrasting assumptions about generativity have succeeded in getting some degree of simple syntactic organization to evolve in a dynamic system shows (a) that both approaches (rules and stored chunks) are compatible with some of the most basic facts of language organization, and (b) that computational evolutionary models have a long way to go in complexity before they can begin to shed light on such issues.

In the model of JB1, the agents are implemented as trainable recurrent neural nets, in which configurations in the output layer are taken to be vectors representing meanings, and the input layer is used for coding successive ‘phonetic’ characters of utterances. One might, perhaps, argue that the rest of the apparatus, the configurations in the hidden and context layers and the weights of all the connections, can then be interpreted as neither semantic nor phonetic, and hence must be ‘syntactic’ in nature. This would be a spurious, even silly, argument.

Syntax in the sense defined involves **compositionality**, the principle that the meaning of a string of symbols is a function of the meanings of the constituent symbols. The evolved systems on which the models discussed here converge all exhibit compositionality. This is achieved in more or less stipulative ways. To introduce the ways in which this compositional relationship between strings and complex meanings emerges, we need first to look at the ways in which these models represent complex meanings.

#### ***1.2.4 Representation and mapping of meanings***

Computers can only manipulate symbols; the human users of computer programs interpret their inputs and outputs semantically, assigning the symbols significance outside the symbolic system. The models of language evolution discussed here adopt sets of symbolic representations which are designated “semantic” or “meanings”. These representations are typically structured according to very simple wellformedness rules

borrowed (uncritically) from such sources as classical predicate logic and various versions of generative grammar. Thus the semantic representations incorporated into the evolutionary models already have a syntax, in the sense of having different classes of terms and strict rules governing the combination and distribution of these terms.

The emergent language systems in these works all have sets of strings which can be analyzed into meaningful substrings, where the meanings of the substrings combine to yield an appropriate meaning for the whole string. To what extent do the emergent syntaxes of the stringsets on which these models converge echo the pre-specified syntaxes of the given sets of meanings? In this respect, again, the SK1, SK2 and JH models contrast as a group with the JB2 model; JB1 is similar to SK1, SK2, JH, but has some interesting characteristics deriving from its neural net implementation.

In SK1, SK2 and JH, in general, where the pre-specified meaning representations contain  $N$  classes of term (e.g. ‘predicates’ and ‘referents’), the stringsets of the emergent languages will also contain  $N$  distributional classes of term (which one can choose to interpret as, for example, ‘verbs’ and ‘nouns’); and the emergent mapping between semantic classes and surface syntactic classes is typically one-to-one. These models assume standard predicate-argument relations in their semantic representations.

SK1 has semantic representations such as [**ag-john, pt-mary, pr-love**], for which a handy mnemonic is the English *John loves Mary*. Each such representation is a triple of attribute-value pairs, in which exactly one attribute (or slot) is always **pr**, suggesting ‘Predicate’, another is always **ag**, suggesting ‘Agent’, and the third is always **pt**, suggesting ‘Patient’. The Predicate slot is always filled by a term drawn from one set (the set of ‘actions’), and the Agent and Patient slots are always filled by terms from a second, distinct set (the set of ‘objects’). Again, this model contains a semantic component into which pre-specified regular and structured representations are built. The emergent grammar groups the arbitrary syllables of the ‘phonetic’ level of the language into two syntactic classes, one for the syllables expressing the ‘actions’ and the other for the syllables expressing the ‘objects’. SK2 and JH similarly converge on grammars in which the distributional classes of the ‘phonetic’ elements directly mirror distributional classes of terms in the semantic representations. Furthermore, in SK2 and JH, in which recursive embedding of propositions as arguments of predicates is allowed,

the structures of the emergent languages all directly reflect this embedding in what can be interpreted as grammatical clause-subordination. Thus, these models converge on ways of topographically mapping semantic form in strings of ‘phonetic’ symbols, taking advantage of the availability of the generalizing facilities inherent in their specified grammatical formalisms. In a quite clear sense, the emergent languages in JH and SK2 simply mirror pre-specified hierarchical semantic structure.

The emergent languages in SK2 and JB2, while relating to semantic structure in quite different ways, have a feature in common, namely that they, unlike SK1 and JH, contain meaningless substrings. In SK2, for example, a language emerges from one run in which the string `stlwrkazqpf d` means *Pete knows that John loves Mary*. Glossing this string in the way familiar to linguists, one can see that only some parts of this string have any semantic interpretation.

stlw	r	k	a	z	gp	f	d
	love	Mary	John	know		Pete	

Here, the substrings `stlw`, `gp`, and `d` are not lexical items in the language, i.e. they are not interpreted as any semantic term. These substrings are simply specified as constants in the relevant grammatical rules; they are obligatory grammatical parts of phrasal or sentential structures in the emergent language, without any obvious function. To the extent that they contain such non-lexical items, the emergent syntaxes in SK2 diverge from the prespecified syntax of the semantic representations.

In JB2, there are also empty strings. An example from an emergent language in this model can be glossed as follows.

da	iwa	ke	noz	sa	pay	ke
flee	snake		bite	rat	wave	

The whole string `da iwa ke noz sa pay ke` could be translated into English as something like *A fleeing snake bit a waving rat*. The substring `ke` has no interpretation as any of the predicates in the semantic representation, and is thus contentless. `ke` is never used by itself. Its use in an exemplar is what makes a phrase containing it be interpreted correctly. Batali suggests that this `ke` might serve some function in the language as a marker of a phrasal boundary. If such suggestions can be sustained, it would be an important step toward accounting for the rise in natural languages of function words, such as determiners and complementizers (as opposed to content words like nouns, verbs and

adjectives).

Another example of an empty word in JB2 is seen in a case where the emergent language has alternative word orders for the same meaning, reminding one of an active/passive alternation. A parallel quasi-English example would be *Cat dog chase* versus *Dog foo cat chase*, both meaning that the cat chased the dog, and where *foo* might be interpreted as a passive marker. One wonders whether, in a long run of JB2's model, such alternative ways of expressing the same meaning would survive, as there seems to be no reason why one form should not eventually oust the other, just as lexical synonyms tend to be eliminated.

In JB2, the structure of the emergent languages does not mirror the pre-specified semantic representations as obviously as in SK1, JH and SK2. JB2's semantic representations are 'flat' unordered sets of elementary propositions, such as

*(waved x) (rat x) (fled y) (snake y) (bit x y)*

The  $x$  and  $y$  are treated as variables, not constants. The structures assigned to strings expressing such meanings have two properties absent from the semantic representations, linear ordering and (binary) hierarchical structure. The imposition of linear ordering was also a feature of SK1, JH and SK2, but the emergence of semantically unmotivated binary hierarchical structure over strings is specific to JB2 and the fourth experiment in JH; I will focus on JB2. This binary structure arises from JB2's learning and production algorithms, which build in the principle that complex exemplars have binary structure, and can be plugged into each other to form new structures.

The deep binary hierarchical structures which emerge in JB2 tend to have a linguistically interesting property which Batali calls 'partitioning'. There is a tendency for all words denoting 1-place predicates with a common argument to form a continuous substring, thus beginning to resemble slightly complex noun phrases in natural languages. These phrases are often separated by a word denoting a 2-place predicate, with the result that a string of up to 8 words can sometimes easily be assigned an SVO or OVS phrasal structure. Whereas in JH, SK1 and SK2, the arguments of predicates, which semantically are individual constants, come to be expressed as one-word proper names, in JB2's semantics there are no individual constants, and so no proper names emerge. But the emergence of structures in which 1-place predicate words are grouped together suggests the beginnings of a class of phrasal

structures resembling natural noun phrases. At present, JB2's emergent groupings of words lack some essential features of natural noun phrases. In particular, they do not have a distinctive head word which denotes an object, rather than a state or action, i.e. they have no clear head noun.

The actual implementation of semantic representations is not necessarily directly symbolic. In particular, the neural net representation of JB1 contrasts with the overtly symbolic representations of SK1, SK2, JH and JB2. This potentially makes for an interesting difference in the kinds of emergent language one gets from these models, as I will explain.

In JB1, the agents are implemented as neural nets whose output layers encode semantic representations as bit patterns. The output layer of each net is a set of nodes, partitioned into two subsets. Batali interprets one subset as corresponding to one-place predicates, and the other subset as corresponding to arguments ('referents'). For illustrative purposes, Batali assigns labels such as *happy, sad, ...* to the 'predicates' and pronoun-like labels such as *me, you, ...* to the 'arguments'. Thus a particular setting of the output layer might be interpreted as the predicate logic formula *HAPPY(YOU)*. Batali is careful to point out that this is merely a suggestive interpretation.

In a given setting of the output layer of one of JB1's neural net agents, exactly three of the six designated 'predicate' nodes are set to 1, with the rest set to 0. The four nodes encoding 'referent' information are each a binary digit encoding the presence or absence of some feature, such as **plural** or **inclusion of speaker**. Within each designated subset of nodes, certain combinations do not occur, but there are no restrictions on the distribution of 1s and 0s between these subsets. The system thus contains a level into which controlled representations have been built. It is this severely constrained regularity in one component of the model to which the emergent patterning of the other layer of the neural net agents adapts. Essentially, JB1's evolving population of neural nets finds a set of strings (from a pre-specified vocabulary, also coded as a bit pattern in the opposite layer of a net) which maps naturally onto the semantic representations.

The production algorithm of JB1 emits one character at a time, at each step building a string that gets progressively closer to the whole intended meaning. As this meaning is distributed across a vector of 10 bits, it is possible that the first couple of characters emitted will get sufficiently close to the 'predicate' part of the whole meaning, so that



the choice of the next character is more effectively directed at beginning to approach the ‘argument’ part of the meaning. In this case, one will get discontinuous substrings denoting predicates. And in fact, some such ‘discontinuous words’ do appear in the emergent language of JB1.

Without going into details, I also suggest that the production algorithm of JB1, combined with its distributed representations of meaning, is likely (a) to correlate shorter strings with more distinctive meanings, and (b) to produce a kind of sound-symbolism, in which parts of strings are correlated with classes of similar meanings.

The symbolic models, SK1, SK2, JH and JB2 have an advantage in semantic representation over the JB1 neural net model. Representing a meaning as a triple of three symbolic slots, of which two can be filled by terms from the same set enables one to represent the same entity (e.g. John) as playing either the Agent or the Patient role in a meaning. This is not possible in a neural net encoding such as JB1’s. A model such as JB1’s cannot ‘recognize’ that a configuration of 1s and 0s in one partition of its output layer is to be accorded the same ‘value’ as an identical configuration in another partition of the layer. If one attempted to extend the coverage of JB1 from intransitive to transitive verbs, the model could at best converge on a system in which there was one set of words allocated to Agents and another set of words allocated to Patients, with no recognition of the fact that the same entities could fulfil either role. That is, the inbuilt patterning which leads the actual JB1 model to evolve a set of Predicate words and a distinct set of Referent words would also lead a model extended to 2-place predicates to evolve a set of Agent-Referent words and a distinct set of Patient-Referent words, with nothing corresponding to any recognition of the intended co-referentiality of these words.

### ***1.2.5 Invention and production algorithms***

One focus of E/I models is the question of how language systems can arise from nothing. A child born with a fully modern L.A.D. into an environment in which no language behaviour exists will not develop a full language. But creolization studies (e.g. Kegl, Senghas & Coppola, 1998; Senghas, 1997) show that children can invent new forms which go beyond any data they observe, and a complex language system can emerge in a community in a relatively short time. A degree of inventiveness must be part of the picture of the rise of language. ‘Invention’ here

should not be construed in the same way as the modern invention, by extraordinary individuals, of complex devices which many other people cannot understand. The invention involved in E/I models is something which it is assumed all individuals are capable of, but which is typically only invoked when an individual ‘needs’ to express some meaning for which it has not acquired a form. Invention is treated by all E/I models as an essentially random process, constrained by the in-built assumptions. The way in which invention is modelled, in particular the degree to which the invention is guided by built-in principles of language structure, has an important effect on the speed of convergence on a generalizable, coordinated language system, and, of course, on the eventual shape of the emergent system itself.

In all the models surveyed here, the act of invention is closely linked with the act of producing an utterance. In the neural net model JB1, in fact, the processes of invention and normal utterance production cannot be separated. JB1’s agents are recurrent neural nets, whose inputs are strings of characters and whose outputs encode meanings. Such feedforward networks are unidirectional — they cannot be reversed to model language production with meanings as input and strings of characters as output. Batali resorts to an ingenious way of modelling the production of utterances, by testing each character in the given alphabet<sup>6</sup> to see which character would move the speakers’s own neural net, given its current weightings, closest to the desired meaning. As the input of any arbitrary character will always have some effect on the net, there is, in the JB1 model, no concept of a speaker simply not having a way of expressing a particular meaning. Thus, a separate mechanism of invention, as distinct from the normal production of utterances on the basis of acquired internal representations, is not postulated in JB1. The element of randomness in invention in JB1 is present in the initial random settings of the connection weights of the agents. Although no clear distinction can be made in JB1 between invention and normal production, it is reasonable to interpret the utterances of agents early in a simulation, before they have been trained to any extent by observing the character-input/meaning-output pairings of other agents, as being more like the outcomes of invention, and the utterances produced later in a simulation, after a good deal of learning has happened, as being less inventive and more like the normal production of an agent guided by an

<sup>6</sup> This is only practicable because the given alphabet is so small — just 4 characters.

acquired system.

In all the symbolic models (SK1, SK2, JH, JB2) a clear distinction is implemented between invention and normal production. The important dimension here, which has a significant effect on the emergent systems, is the relationship between invention of forms for atomic meanings and that for complex meanings. An issue arises concerning the extent to which compositionality is built into the system via the invention algorithm (*inter alia*), as opposed to emerging, unprogrammed, from the dynamics inherent in an E/I model. In this respect, there are several clear differences between the SK models and JH. An assumption of compositionality of meaning is clearly built into JH, whereas it is built into the SK models to a much lesser degree, and arguably not at all. I will briefly contrast SK2 and JH in this respect.

Agents, regardless of whether they have any ‘linguistic’ knowledge, are prompted to express meanings. A factor which affects at least the speed, and possibly the converged-upon outcomes of simulations, is the nature of the semantic units which can serve as prompting meanings. In SK1 and SK2, only whole propositions are used as prompting meanings. In SK1, there are no complex propositions, and so all prompting meanings are simple propositions of the *LOVE(JOHN,MARY)* variety. In SK2, the prompting meanings are sometimes simple, and sometimes complex propositions, as in *SAY(MARY,LOVE(JOHN,FIONA))*. In JH, by contrast, the prompting meanings may also be any proper part of a proposition, such as a single predicate or a single individual constant. (It is thus assumed that there may be acts of pure reference, with no predication, and also acts of predication with unexpressed arguments.) This has the effect that learners may be exposed to, and learn from, atomic meaning-form pairs. In the SK models, on the other hand, learners are not exposed to atomic meaning-form pairs. This makes for a difference in the ways in which the emergent language is gradually built up during a simulation. In JH, the lexical items tend to emerge early, and the forms for more complex meanings are later constructed synthetically from them. In the SK simulations, initially unanalyzeable invented strings for complex meanings are only later analyzed by language inducers into substrings to which simpler meanings are assigned.

(The JH and SK models can be taken to imply quite different evolutionary routes from a single-word stage of language, like a Bickertonian protolanguage, to multi-word systems. The route implied by JH is synthetic, with the early, primitive forms bearing simple meanings and

becoming the atoms of the later more complex system. The route implied by SK is analytic, with the early, primitive forms bearing complex meanings, and being subsequently broken down into smaller parts which become the atoms of the later more complex system.)

In the symbolic models, given a full grammar, an agent follows the grammar to produce a form for the prompting meaning. Given no grammar at all, the agent invents a form. Given a partial grammar, an agent produces a string for the prompting meaning that is partly generated by the grammar and partly invented. In JH, the agents invent new forms for hitherto inexpressible meanings and induce rules from observed behaviour in ways which quite directly follow the given semantic structure. For example, if an agent in JH knows words for the argument term *FIONA* and the predicate term *SING*, but, as yet, no grammatical rule for combining these words, then if prompted to express the proposition *SING(FIONA)*, the agent is ‘intelligent’ enough to know that the required expression should contain the words for *FIONA* and *SING*; what it will not know, and therefore have to invent, is the order in which these words are to be arranged. JH’s agents, then, are credited with knowing that expressions for complex meanings should be composed of parts which express the simpler components of those meanings; in a sense, compositionality is built into the model. The emergent languages, not surprisingly, have words for each semantic term, and impose a linear ordering on these terms within propositions. In SK2, as in SK1, the invention and induction algorithms are less ‘intelligent’ than in JH and do not obviously build in a principle of compositionality. In fact, Kirby claims that compositionality **emerges** in this model without being deliberately coded in.

In both JH and SK2, at intermediate stages in a run, utterances are produced which are partly rule-generated and partly invented. But there is a difference which may be crucial. In JH, an invented form is always a form invented to express the form for some well-defined constituent of the hierarchical semantic representation, and such a form also ends up as a proper constituent of the complex forms in the evolved language. In SK2, an invented form also corresponds to a well-defined constituent of the hierarchical semantic representation (often, especially in the early phase, the whole proposition to be expressed). But the invented strings in SK2 do not necessarily end up (though they may) as proper constituents of complex forms in the evolved language.

In SK2, the invented parts of an utterance are unstructured sequences

of characters, which may later be ‘made sense of’ by the induction algorithm of other agents. Furthermore, in JH, the invention process can be called recursively, so that invented strings can contain invented substrings; and the relation of an invented substring to a larger invented string mirrors exactly the hierarchical structure of the semantic representation. Clearly, JH attributes to its agents more ‘awareness’ of the structure of semantic representations than the SK models do. This may or may not reflect a plausible assumption about human-like creatures. What is notable is that the SK models converge on stringsets which are systematically mapped onto pre-existing semantic structures without such explicit direction from the invention algorithm.

The branching structures of the emergent languages in SK1 and SK2 can differ significantly from those in JH. The branching structure of the emergent language in JH is constrained to be exactly the same as that of the pre-defined semantic representations, because the invention algorithm follows the semantic tree structures. All that the JH model does, in effect, is invent lexical forms for the semantic atoms and impose an invented linear order on them, within the pre-defined hierarchical structure. Consequently, in JH, if the semantic structure is binary branching, its emergent linguistic form will also be binary branching; if the semantic structure is three-way branching, its emergent linguistic form will also be three-way branching, and so on<sup>7</sup>.

In SK1 and SK2, on the other hand, since the invention algorithm is not guided to invent chunks exactly corresponding to proper constituents of the hierarchical semantic structure, the emergent phrase structures are often many-ways branching (sometimes because of the inclusion of ‘meaningless’ elements, as discussed above). It is possible, in the SK models, for example, to get emergent VP structures bracketing together the forms for a predicate and just one of its arguments. The branching structures defined by the emergent grammars in the SK models can be quite heterogeneous.

In JB2, the production algorithm imposes a binary branching structure on the building blocks (the exemplars) of the emergent language. All emergent complex structures over strings are, by definition, binary branching. It is this binary branching structure, rather than any structure in the semantic representations (which are flat), which guides JB2’s

<sup>7</sup> All the generalizations about JH in this paragraph are subject to the reservation “except for the fourth experiment”.

invention procedure. In JB2, when a new string is invented, it is either a form for a whole meaning or for a proper subpart of an existing binary branching structure. The invention algorithm in JB2, then, like that in JH, also builds in to the model an assumption of compositionality.

### 1.2.6 *Induction algorithms*

Grammar-induction was implemented in strikingly different ways in these models. But all models had some features in common. The common ground exists in their treatment of new examples early in simulations at stages when agents have relatively little stored linguistic knowledge. At early stages in all models, the mode of learning was what Langley would classify as ‘incremental’.

A ... distinction holds for learning algorithms, which can either process many training instances at once, in a *nonincremental* manner, or handle them one at a time, in an *incremental* fashion.

*Langley (1996:19)*

Given a newly presented meaning-form pair, as yet unanalyzeable by any of the agent’s rules or exemplars, all symbolic models simply store this meaning-form pair. This happens even if the meaning is quite complex, with the result that agents early in a simulation tend to have vocabularies of holistic expressions idiosyncratically linked to a random range of meanings.

For later stages in a simulation, by which time agents typically have internalized large sets of rules or exemplars, one can differentiate the models along a scale according to how much internal rearrangement of an agent’s previously stored information takes place. On this scale, JH and JB2 are at one end, and the SK models are at the other. Learners in JH and JB2 respond to a presented meaning-form pair by attempting a parsing or analysis of it in terms of their existing grammars. In JH and JB2 the minimum addition necessary to enable a grammar to analyze the given meaning-form pair is made to the grammar. In JH and JB2, the reorganization of an agent’s grammar specifically in response to a given observation can result in the addition of a new piece of grammatical information, but never in the deletion of an existing rule or in any change to the substance of an existing rule; existing rules may, however, be ‘demoted’ in various ways, so that they become less likely to be used in later learning and production. The SK models, on the other hand, after each presentation of a meaning-form pair, firstly take in this

meaning form-pair ‘raw’, without analysis, but then undertake an exercise of rationalization of **the whole existing grammar** (including the new example), with a view to seeking coincidental similarities between rules, and collapsing them where generalizations over them are possible. This collapsing of rules involves introducing variables where there were previously constants. SK’s inducer is like an obsessively tidy librarian, who at every opportunity (e.g. after each book is returned) tries to rearrange the items in his whole storeplace in the most economical and general way. The JH and JB2 models, by contrast, take a less global approach to the maintenance of agents’ grammars.

This dimension, on which SK differs from JH and JB2, is not exactly the dimension of incremental versus nonincremental learning as characterized by Langley(1996). All the models are strictly incremental, in that they process one datum at a time. In SK, strictly speaking, no revisiting of previous data actually happens, but there is substantial revisiting and reprocessing of the internal representations directly caused by the earlier data.

On another dimension, the induction mechanisms of the JH and SK models fall together, with JB2 differing. This is in the degree of generalization which an inducer does. As mentioned earlier, in the section on syntactic representation, JB2 essentially does not make generalizations, by inducing rules containing variables, but rather stores whole exemplars, which may have much in common, but are not explicitly generalized over. In JB2, the work done by a generalizing inducer is in a sense done by the production algorithm in finding the least costly way of combining stored exemplars. SK and JH induce general rules, while JB2 does not.

Although both SK and JH induce general rules, their methods are radically different. In SK, the generalization is done by a search involving comparison of all pairs of rules. In JH, the inducer can infer a new lexical rule or general constituent-ordering rule simply by exposure to a single example. JH thus implements one-exposure learning, an extreme form of generalization. In principle, this is no different from accumulating a larger set of examples with some property in common, and then making the appropriate generalization when the number of examples reaches some critical number; in JH this critical number is simply 1. A constraint on this one-exposure learning in JH was that only one new rule could be acquired from any given example.

The topics of invention and induction are linked by the question of whether agents learn from their own inventions. In SK1 and SK2, an agent learns from its own inventions/productions. In JH and JB2, this is not the case. Learning from ones own inventions speeds up the social coordination of the acquired system, as this makes a particular agent's productions more self-consistent. In JH and JB2, where agents do not learn from their own productions, successive invented forms for the same meaning are not constrained to be the same; the inventor/speaker does not listen to itself. The feature of 'self-teaching' is, at least in some models, dispensable without any effect on whether the model eventually converges. Whether self-teaching is actually dispensable in all models is not clear.

Finally, the assumed relationship between production/invention and reception/induction appears to differ in the models surveyed, along a dimension that might be glossed as 'explicit guidance by *a priori* unifying principles'. In JH and JB2, on the one hand, the algorithms for production and learning are explicitly constructed around assumed common principles defining the possible mappings between meanings and strings. The production and the reception algorithms are both informed by a knowledge of the same pre-defined possibility space of meaning-form mappings, and are both essentially, *mutatis mutandis*, searches of this space. In JH and JB2, there is a clear sense in which the response of a learner to a particular example (by acquiring a some rule(s) or exemplar(s)) can retrace, in reverse, and given enough shared linguistic knowledge between speaker and hearer, a similar route to that by which the example was produced. In the SK models, on the other hand, the production/invention and induction algorithms are defined independently of each other, and emphasis is not laid on their being built around a search of the same space of meaning-form mappings.

But this difference between models is more apparent than real. In all E/I models, the workings of production/invention and reception/induction mesh with each other. In SK, both production and induction algorithms are based on the assumption that the linguistic knowledge of a speaker is represented as a semantically augmented context free phrase structure grammar, which in fact, along with the semantic and 'phonetic' representations, does define the space of possible meaning-form mappings. It would be possible, although it is not done in the models surveyed here, to define production/invention and reception/induction in terms of different assumptions about the space of possible mappings between



form and meaning. That is, sometimes a speaker might invent a form according to a type of generalization that is actually unlearnable by a hearer/learner; and conversely there might be some learnable generalizations of which examples cannot be systematically produced by any speaker. In such a case, one might expect an emergent language to fall in the intersection of the two spaces.

### **1.2.7 Bottlenecks**

Of the models surveyed here, JB1 alone implemented a very weak semantic bottleneck (90% of meanings), and this factor probably influenced the outcome. This follows from the conflation, in JB1, of production and invention, which in turn follows from the neural net implementation. A neural net, trained or untrained, will always respond to some input by giving some output. There is no distinction, in the neural net implementation, between meanings which an agent knows how to express, and those which it doesn't know how to express. Given any meaning, the JB1 algorithm will find the best (often, rather, the least bad) form to express it. A trained net will tend often to converge on a number of distinct attractor states that is no greater than the number of meaning-form pairs presented to it in training. Thus if any meaning-form pair is omitted from training, the net will sometimes respond to the form by conflating its meaning with that of some other which it 'knows'. Applying a semantic bottleneck, that is withholding some meanings from the training schedules, can result in under-differentiation of the meaning-space. How strong a neural net model such as JB1 can 'tolerate', while still leading to an emergent system distinguishing all possible meanings is an open question.

All the other models, with symbolic (rather than neural net) architectures and dynamics, did implement a semantic bottleneck, and this, paradoxically, was vital to their outcomes. In a model with a semantic bottleneck, an agent is always likely to be prompted to express some particular complex meaning which did not form part of its learning experience. In this situation, one of two things may happen. The agent may have generalized from the meaning-form pairs which comprised its learning experience some rule which does cover the prompting meaning, and it will then apply this general rule to produce the required form. Any such general rule will use one or more variables ranging over subparts of the complex meaning and their possible forms, and an appropriate

form is produced. The other possibility is that the agent will not have acquired an appropriate general rule, and will thus resort to inventing a novel form for this particular complex meaning. Thus the omission of particular meanings from the learning experience of one agent causes in its subsequent production behaviour either the invention of idiosyncratic forms for these meanings or the application of general rules to produce forms similar in shape to forms with related meanings. A general rule acquired by one agent covering  $N$  different form-meaning pairs will be  $N$  times more likely to be represented in the learning experience of some other agent learning from it than a one-off idiosyncratic form-meaning pair. Thus, models will converge on behaviour conforming to general rules. This mechanism is at the heart of symbolic E/I models.

If a symbolic E/I model did not implement a semantic bottleneck (which is only possible with a finite semantic space), then no agent would ever be forced to generalize beyond its learning experience, and the aboriginal first-invented forms for each meaning would simply be re-used and relearned by each generation.

As mentioned earlier, all the models, very naturally, implemented a production bottleneck, so that, when prompted with a particular meaning, an agent acting as speaker would only ever use a subset of its possible forms for that meaning. Both JH and JB2 implement mechanisms which promote the use of commonly experienced meaning-form pairings, thus creating a positive feedback loop. JB2's mechanism is quite complex, involving searching through exemplars related to a given meaning-form pair in the agent's existing grammar, and adjusting their cost. JH's mechanism is brutally simple; acquired rules are stored in order of acquisition, and earlier stored rules are always favoured in use over later acquired rules. As earlier acquired rules tend to be those more commonly used by other agents in the examples presented to the learner. Rules at the bottom of an agent's list may in fact never be used, and thus do not give rise to examples from which other agents can learn. SK1 kept a numerical count of the examples used to induce particular rules, and given a choice in the production task between two rules, used the rule with the highest empirical justification. The obvious effect of these devices is to reduce synonymy.

A further kind of bottleneck is also, but less often, as it happens, directly implemented in the models surveyed here. In language acquisition, one should distinguish between input and intake. That is, not all the meaning-utterance pairs presented to a child as data may actu-

ally be taken in and used in language acquisition. The child's 'trigger experience' can be a subset of the primary linguistic data to which she is exposed. This could be modelled by placing a selective filter (alias bottleneck) on the language acquisition device. This could be labelled an '**intake bottleneck**'. (See Kirby (1999) for extensive discussion of the effects of intake bottlenecks on diachronic language drift.) The SK models had a kind of intake bottleneck. The SK induction algorithms simply ignored any meaning-form pair for which the agent's rules already assigned some meaning to the presented form, regardless of whether the presented meaning was the one its existing grammar assigned to the presented meaning. In effect, this prevents homonymy from arising in the emergent system.

Models without any kind of bottleneck produce no interesting kind of linguistic evolution. It is also clear that, in the actual transmission of human languages across generations, there are huge bottleneck effects. This is simply to reiterate the axiom that the grammar of a language is massively underdetermined by the observed data.

The size of the bottleneck is an important variable. The size of a bottleneck determines how much data a learner is exposed to during its learning period. If the bottleneck is too small, the learner is simply not given enough data from which to generalize, and no interesting syntactic system can emerge. If the size of the bottleneck is too large, the learner is given ample opportunity to internalize a set of non-general statements giving somewhat adequate coverage of the whole language as defined by the internalized grammars of the previous generation, whose behaviour it has observed; in such a case, internalized general rules are slow to emerge. Where a language is in principle infinite, through recursion, no finite amount of data can exhaustively exemplify the whole language, and there is thus pressure on general syntactic rules to emerge in any case, but E/I simulations in which the bottleneck is set rather large are known to take longer to converge on syntactic systems than the experimenter has time for, running for perhaps millions of simulated generations without convergence. (The idea of the 'size' of a semantic bottleneck is in fact only applicable to 'multi-generational' models — see next subsection.)

### 1.2.8 *Population dynamics*

In the models surveyed here, the agents in the simulated populations interact in rather different ways. We can distinguish two broad types, which I will label **multi-generational** and **uni-generational**. Within these types there are also some differences between models.

JH, SK1 and SK2 are multi-generational models. In such models, agents are periodically removed from the population, and their grammars die with them. Only the effect that their behaviour has on a cohort of learners lives on. When an agent dies, it is replaced with a 'newborn' agent without any internalized grammar, but only the innate capacity to induce a grammar from observed behaviour. In the multi-generational models discussed here, the number of agents in the population at any given time is kept constant.

(Conceivably, simulations could permit the population size to expand and contract historically, with correspondingly varying proportions of learners and adult performers. In a period of population expansion after a period of contraction, the proportion of young people in the population increases. In more sophisticated models than these, such details might give rise to some phenomena of theoretical interest, with periods of linguistic simplification correlating with periods with a high proportion of learners in the population. See Johansson (1997) for some detailed work along these lines of thought.)

SK1 reports a simulation with a population of 10, in which the single most recently introduced individual is designated as the learner. In SK1, there is the added complication of some simulated spatial organization of the population. The individuals each occupy a location in a notional two-dimensional space, so that it is possible to identify an individual's neighbours. The learner only observes, and learns from, the behaviour of two of its immediate neighbours in the population. None of the other models discussed here implemented spatial organization, and while it may have had some acceleratory effect on the outcomes in SK1, spatial organization does not seem to be vital to the emergence of syntax in such models. (See Di Paolo (1999, 1998, 1997); Oliphant (1997, 1996) for other related work on the effects of spatial organization in the evolution of language.) In fact, the typical size of a population in these simulations is so small as to make spatial differentiation unrealistic. The JH model used a fixed population size of 5, including one designated learner, with no spatial organization.

The minimum population size which will allow for an essential feature

of E/I models, namely the acquisition of a grammar by an agent on the basis of observation of at least one other agent, is 2. SK2 works with this minimum population size: at any one time, there is just one speaker whose behaviour is being observed, and just one learner. After a certain time, the ‘adult’ disappears, the former learner becomes a speaker, and a new learner is introduced. There is no overlap of generations in this version of the model.

A population size of just 2, with no overlap of generations, as in SK2, may seem an overly drastic simplification. But in fact it usefully eliminates one factor from the evolutionary scene. In all the reported work, except SK2, at least part of what is going on in a simulation is the social coordination of co-eval individuals. Given an adult population of more than 1, there will be, especially at the beginning of a run, when invention of forms is still in full swing, a variety of different forms for the same meaning. Part of what happens in these simulations is simply standardization of usage between individuals. But this is not the real focus of these studies. The more interesting phenomenon is the evolutionary transition into syntax, which is not a matter of coordination among individuals, but a matter of how successive single individuals organize their mental representations of their language. A simulation with only one individual learner and one individual transmitter per generation simply avoids the work of having to get the population coordinated, as well as developing syntax.

Simulations in the multi-generational models needed to run for varying numbers of generations before interesting results emerged. A run reported in SK1 ran for 500 turnovers of the whole population of 10, i.e. for 5000 births. A simulation reported in SK2 took almost 8000 births before converging on an elegant syntax; Kirby (personal communication) tells me that other runs of the SK2 model converged much faster, sometimes in as little as 30 births. The simulations in JH typically achieved results very quickly, usually less than 100 births; this speedy convergence is due to the unrealistically great generalizing power attributed to learners and inventors in JH. In fact, the issue of ‘time’ to convergence is difficult, if not impossible, to interpret in any empirically enlightening way, given the high degree of idealization and simplification of human communities and minds found in all these models.

The JB1 and JB2 models are uni-generational. In such simulations, the population comprises the same set of individuals for the whole of a run. Changes take place within these agents, as they learn from the be-

haviour of their companions, internalizing grammars. These are models of stable populations in which all individuals learn by listening to the others. As the agents acquire more and more knowledge of the community language (as the language itself simultaneously begins to form), they change. Just as a real person can be, in some sense, a different person from the one he was a few decades earlier in life (for example by having experienced more, or forgotten some of what he knew before), so these simulated agents ‘become different people’. But they are not different people in the radical sense of having been ‘biologically’ conceived and introduced with zero knowledge into the population, as in multi-generational models.

The question arises whether uni-generational models such as JB1 and JB2 incorporate a bottleneck in the same sense as multi-generational models. With uni-generational models, one may perhaps say that the metaphor of a bottleneck is less appropriate, as representations of the language are not passed from one generation to the next via a small set of examples. But nevertheless, a kind of bottleneck effect is present, as we have defined it, because learners’ internal representations are induced from limited numbers of examples of the behaviour of other agents. Two further factors are characteristic of the JB1 and JB2 models; these are **decay of unused internal representations** and a **cost metric on internal representations**. These factors may contribute to the result of convergence on syntactic systems, compensating for the lack of the specific kind of bottleneck found in multi-generational models.

JB1 implements agents as neural nets. The agents ‘talk to each other’ and thereby train each other in the emerging community language. As this training goes on and on, the weights in the nets are constantly being readjusted in response to the most recent training data, and any residual effect of data presented earlier, to the extent that it is incompatible with later data, is superseded.

The JB2 model is implemented symbolically (not in neural nets), and there is an explicit pruning procedure by which any internalized statements that have not been used for a certain number of episodes are deleted from an agent’s memory. Clearly, one could set up a uni-generational simulation with this kind of grammar decay in such a way that it was effectively a notational variant of a multi-generational model. One could, for example, partition the population into two halves, and decree that every even hundred episodes (i.e. at 200, 400, 600 , . . . episodes), one half of the population forgets everything it has learnt,

and starts to learn anew by listening to individuals from the other half of the population; this other half of the population would similarly lose all its knowledge of the language every odd hundred episodes (at 100, 300, 500, ... episodes). Forgetting everything is like being born again. Partial forgetting, as in JB1 and JB2, is partially like being born again, and to this extent, there is some effect similar to the intergenerational bottleneck effect seen in SK1, SK2 and JH.

Pruning of rules is not exactly equivalent to killing off agents. When JB2's exemplars are pruned, it is because they are not part of the agent's active repertoire. But when an agent dies, **all** of its exemplars (or rules) are eliminated. If similar rules or exemplars are possessed by surviving members of the population, the disruption to the continuous evolution and transmission of the community's language might be negligible.

Here a provocative question arises. Would a unigenerational model with a bare minimum of one agent learning from itself, with forgetting of infrequently used rules, converge, like the models surveyed here, on a system with some incipient syntax? The experiment remains to be tried. Obviously, it would not model an actual human historico-cultural process so closely as the models surveyed here. But if it did produce an emergent system of similar interest, this would reveal a new lower bound on the conditions under which an E/I model could produce language emergence. Even such a bare model would still have the essential ingredients of an E/I model, namely the constant cycling of information through agent-internal representations and external behaviour.

**Postscript:** John Batali (personal communication) tells me that he has conducted single-agent simulations just as described here, mostly with rapidly converging results. And Timm Euler, in a dissertation at Edinburgh University, has implemented a single-agent, 'talking-to-onself' version of SK2, with decay of little-used rules. Euler's model also produces evolutionary convergence on syntactic systems, with some interesting differences from SK2.

### 1.3 Methodology: What IS a language?

The workings of the models surveyed here raise a fundamental question about any model of the dynamic historical interaction between individuals' mental lexicons/grammars and their production behaviour. The overt speaking behaviour of any agent at many stages in a simulation

will not give a faithful picture of the totality of its acquired meaning-form pairings. The expressions actually used will be a subset of those internally represented. The spirit of this research paradigm is that a language is neither just I-Language nor just E-Language, but their dynamic interaction. The conclusion truest to this spirit is that the language over any given time period (say a generation) is a pair, consisting of both the (perhaps heterogeneous) internalized lexicons/grammars of the individuals in the population and the totality of their behaviour. Defining a language in this way is hardly elegant, but (a) it recognizes the essential interdependence of the two phases of language, I-Language and E-Language, and (b) it avoids an arbitrary privileging of one phase over another.

Symbolic computational modellers enjoy a luxury unavailable to empirical grammarians, in that they can directly inspect the grammars of their simulated agents<sup>8</sup>. For generative grammarians, the principal method of accessing speakers' grammars is by asking for their intuitions of the well-formedness of presented examples. In questioning native speakers' grammatical intuitions, typically no distinction is made between what can be called 'active wellformedness' and 'passive wellformedness'. The metatheory of generative grammar holds that a grammar is neutral with respect to production and reception. In fact, however, human speakers will often to respond to presented examples with statements such as "Well, I can understand what it means, and I suppose you would call it grammatical, but I wouldn't put it that way myself", or "It's not actually ungrammatical, but I'd say it differently myself". Such responses indicate that a speaker's productive language behaviour reflects only a subset of the meaning-form correspondences that her internalized grammar can recognize. In the simulations surveyed here, it is very common for an agent's grammar to specify some meaning-form correspondences which the agent would never actually use in production. A definition of the language system of such agents would in some sense be wrong if based solely on their production behaviour; and a definition based only on the form-meaning correspondences characterized by their internal grammars would miss an important difference in the actual **realizability** of these correspondences in the agents' production behaviour. The existence of a production bottleneck in the usual life-cycle of a language obliges us to take a view of a whole dynamic

<sup>8</sup> Modellers working with agents implemented as neural nets are denied this luxury.



language system in terms of both E-language (the productions) and I-language (the internalized grammars).

#### 1.4 Conclusions

Factors which facilitate the emergence of recursive, compositional syntactic systems in E/I models are:

- Pre-defined hierarchical semantic representations. These are present in SK1, SK2 and JH, but not in JB2, which has flat semantic structures.
- An invention (and/or production) algorithm disposed, to a greater or lesser extent, to construct new expressions in conformity with the principle of compositionality. In this respect, the invention algorithm of JH most strongly builds in compositionality, followed closely by the invention/production algorithm of JB2. The invention algorithms in SK1 and SK2 are least biased toward invented forms that automatically conform to the compositionality principle.
- A learning algorithm disposed, to a greater or lesser extent, to internalize rules generalizing over form-meaning mappings in a compositional way. In this respect, the learning procedures of JB2 are least biased to internalize such rules; JB2's JB2agents do not internalize rules at all. JH's learning procedure, except when it is partially disabled, as in the third experiment of JH, is strongly biased to internalize compositional rules wherever possible. The rule-collapsing procedures of the SK learners are also disposed to arrive at general rules incorporating compositionality wherever possible.
- A strong semantic bottleneck effect, by which the information in the evolving system is recycled very frequently through the alternating phases of representation as internal rules (I-Language) and exemplification in utterances (E-Language), and learning takes place on the basis of examples covering a very small subset of the available semantic space. This effect is strongest in the multi-generation models, SK1, SK2 and JH, and least strong in JB2, a uni-generational model.
- A production bottleneck effect, by which certain rules are preferred over others in the production process. All the models discussed incorporate such a production bottleneck, based explicitly or implicitly on the frequency of examples in the learning experience, thus effectively providing a feedback loop favouring rules of greater generality.

It is tempting to imagine a ‘minimal hybrid model’, eclectically put together out of those components of the various models which are the least disposing to the emergence of a recursive compositional system. But it seems clear that an extreme minimal hybrid model would not yield an emergent recursive compositional syntactic language system. In particular, an opportunity (but not a compulsion) for compositional rules to arise must be present in either the invention/production algorithm (as in JB2) or in the induction algorithm (as in SK1 and SK2). Given that either the invention/production component or the induction component of a model must allow for compositional rules to arise, the availability of such rules can nevertheless be impaired or restricted, as in the third experiment of JH.

I end, then, with a bold speculation that a hybrid model stripped down to the following components would yield an emergent recursive compositional syntactic system.

- Flat semantic structures as in JB2.
- An invention algorithm with no bias toward compositional structures, as in the SK models.
- A learning algorithm, such as JH’s, SK1’s or SK2’s, which may, but does not only, induce general rules incorporating a principle of compositionality; the disposition of such a learner to form general compositional rules could be experimentally disabled to various extents, to determine a minimum level necessary for a syntactic system to emerge.
- Uni-generational population dynamics, as in JB2, but with only a single agent, talking to, and learning from, itself, plus some experimentally varied decay and forgetting of rules, giving a weak semantic bottleneck effect.
- The feedback effect of a production bottleneck experimentally weakened by allowing various degrees of randomness in the selection of the rules by which utterances are produced.

Now there’s a set of experiments crying out to be done!

### Bibliography

- Andersen, H. (1973). Abductive and deductive change. *Language* 40, 765–793.
- Batali, John, (1998). Computational simulations of the emergence of grammar. In James R. Hurford, Michael Studdert-Kennedy and Chris Knight (Eds.) *Approaches to the Evolution of Language: Social and cognitive bases*, pp. 405–426. Cambridge University Press.

- Batali, John, (this volume). The negotiation and acquisition of recursive grammars as a result of competition among exemplars.
- Bod, Rens, (1998). *Beyond Grammar: an experience-based theory of language*. Stanford, California: CSU Publications, Center for the Study of Language and Information.
- Bolinger, Dwight (1968). *Aspects of Language*. New York: Harcourt, Brace and World.
- Bolinger, Dwight (1976). Meaning and Memory. *Forum Linguisticum* 1, 1–14.
- Chomsky, Noam (1971/1965). Paper read at the Northeast Conference on the Teaching of Foreign Languages, 1965. Reprinted in J. P. B. Allen and Paul van Buren (Eds.), *Chomsky: Selected Readings*, Oxford: Oxford University Press.
- Clark, R., and I. Roberts (1993). A computational model of language learnability and language change. *Linguistic Inquiry* 24,2, 299–345.
- Coulmas, Florian (1979). On the sociolinguistic relevance of routine formulae. *Journal of Pragmatics* 3, 239–266.
- Coulmas, Florian (1981) (Ed.). *Conversational Routine*. The Hague: Mouton.
- Di Paolo, Ezequiel (1999). *On the Evolutionary and Behavioral Dynamics of Social Coordination: Models and Theoretical Aspects*. D. Phil Thesis, University of Sussex.
- Di Paolo, Ezequiel (1997). Social coordination and spatial organization: Steps towards the evolution of communication. In Phil Husbands and Inman Harvey (Eds.) *Proceedings of the 4th European Conference on Artificial Life, ECAL97*. MIT Press/Bradford Books: Cambridge, Mass.
- Di Paolo, Ezequiel (1998). An investigation into the evolution of communication. *Adaptive Behavior* 6,2, 285–324.
- Gibson, E., and Kenneth, Wexler (1994). Triggers. *Linguistic Inquiry* 25,4, 407–454.
- Hare, M., and Jeff Elman (1995). Learning and morphological change. *Cognition* 56, 61–98.
- Hurford, James R., (in press). Social transmission favours linguistic generalization. In Chris Knight, Michael Studdert-Kennedy and James R Hurford (Eds.) *Approaches to the evolution of language: The emergence of phonology and syntax*. Cambridge: Cambridge University Press.
- Johansson, Christer (1997). *A View from Language: Growth of language in individuals and populations*. Lund University Press.
- Kegl, Judy, Anne Senghas, and M. Coppola (1998). Creation through contact: sign language emergence and sign language change in Nicaragua. In M. DeGraff (Ed.) *Language Creation and Change: creolization, diachrony and development*, Cambridge, Mass: MIT Press.
- Kirby, Simon (1999). *Function, Selection, and Innateness: The emergence of language universals*. Oxford: Oxford University Press.
- Kirby, Simon, (this volume). Learning, bottlenecks and the evolution of recursive syntax.
- Kirby, Simon, (in press). Syntax without natural selection: How compositionality emerges from vocabulary in a population of learners. In Chris Knight, Michael Studdert-Kennedy and James R Hurford (Eds.) *Approaches to the evolution of language: The emergence of phonology and syntax*. Cambridge: Cambridge University Press.
- Langley, Pat (1996). *Elements of Machine Learning*. San Francisco, California: Morgan Kaufmann Publishers, Inc.

- Lightfoot, David (1999). *The Development of Language: Acquisition, change and evolution*. Oxford: Blackwell Publishers.
- Nattinger, J. R. and J. S. DeCarrico (1992). *Lexical Phrases and Language Teaching*. Oxford: Oxford University Press.
- Niyogi, Partha, and Robert Berwick (1997). Evolutionary consequences of language learning. *Linguistics and Philosophy* 20, 697–719.
- Oliphant, M. (1997). *Formal Approaches to Innate and Learned Communication: Laying the foundation for language*. Doctoral dissertation, Department of Cognitive Science, University of California, San Diego.
- Oliphant, M. (1996). The dilemma of saussurean communication. *BioSystems* 37,1-2, 31–38.
- Peters, Ann (1983). *The Units of Language Acquisition*. Cambridge: Cambridge University Press.
- Senghas, Anne (1997). *Children's Contribution to the Birth of Nicaraguan Sign Language*. Ph. D dissertation, MIT.
- Steels, Luc, (1996). Self-organizing vocabularies. In: Chris Langton (Ed.) *Proceedings of Alife V*, Nara, Japan.
- Steels, Luc, (1996). A self-organizing spatial vocabulary. *Artificial Life Journal* 2(3).
- Steels, Luc, (1996). Emergent Adaptive Lexicons. In: Maes, Pattie (Ed.) *Proceedings of the Simulation of Adaptive Behavior Conference*. Cambridge Mass.: MIT Press.
- Steels, Luc, (1997). The Spontaneous Self-organization of an Adaptive Language. In: Muggleton, S. (Ed.) *Machine Intelligence 15*. Oxford: Oxford University Press.
- Steels, Luc, (1998). The origin of linguistic categories. Report, SONY Computational Research Laboratory, Paris.
- VanLancker, D., (1975). Heterogeneity in Speech. *UCLA Working Papers in Phonetics*, 29.
- Vogt, Paul (1998). The evolution of a lexicon and meaning in robotic agents through self-organization. In La Poutri, Han, and Jaap van den Herik (Eds.) *Proceedings of the Netherlands/Belgium Conference on Artificial Intelligence*. Amsterdam: Centrum voor Wiskunde en Informatica, 7–22.
- Widdowson, Henry, (1984). Comments on the implication of interlanguage for language teaching. In A. Davies, C. Cripser and A. P. R. Howatt (Eds.) *Interlanguage*. Edinburgh: Edinburgh University Press.
- Widdowson, Henry, (1990). *Aspects of Language Teaching*. Oxford: Oxford University Press.
- Weinert, Regina, (1995). The role of formulaic language in second language acquisition: A review. *Applied Linguistics* 16,2, 180–204.
- Wray, Alison, (1998). Protolanguage as a holistic system for social interaction. *Language and Communication* 18, 47–67.
- Wray, Alison, (forthcoming). Holistic utterances in protolanguage: the link from primates to humans. To appear in Chris Knight, Michael Studdert-Kennedy and James R Hurford (Eds.) *Approaches to the Evolution of Language: the emergence of phonology and syntax*. Cambridge: Cambridge University Press.
- Yorio, C. (1989). Idiomaticity as an indicator of second language proficiency. In K. Hyltenstam and L. Obler (Eds.) *Bilingualism Across the Lifespan: Aspects of acquisition, maturity and loss*. Cambridge: Cambridge University Press.

---

## index

- adaptation, 5  
analogies, 17  
analytic route to complex language, 28
- Batali, J., 1, 5, 13–27, 29–34, 37–39, 41, 42  
Berwick, R., 4, 5  
Bickerton, D., 27  
binary branching structure, 23, 29  
Bod, R., 20  
Bolinger, D., 17, 19  
bottleneck, 6–9, 11, 38  
bottleneck, intake, 35  
bottleneck, production, 7–10, 34, 41, 42  
bottleneck, semantic, 6, 9, 10, 33, 34, 41  
bottleneck, size of, 35
- case-based learning, 18  
Chomsky, N., 3, 17, 19  
Clark, R., 5  
communication, 5  
compositionality, 2, 11, 12, 17, 20, 21, 27, 28, 30, 41, 42  
connectionism, 11  
Coppola, M., 25  
Coulmas, F., 19  
creolization, 25
- DeCarrico, J.S., 19  
decay, memorial, 38, 39  
Di Paolo, E., 36  
discontinuous substrings, 25
- E-Language, 3, 40, 41  
Elman, J., 4  
emergence, 2, 6  
Euler, T., 39  
evolution, 8  
evolution from no language, 4  
evolution, biological, 4  
evolution, cultural, 4  
exemplars, 17–19, 29, 31  
Expression/Induction models, 2, 7, 39
- formulaicity, 20, 30  
frequency, 10, 34  
function words, 22
- generalization, 16, 18, 31, 33, 34  
Gibson, E., 5  
grammar, acquisition of, 4  
grammaticality judgements, 40
- Hare, M., 4  
homonymy, 9, 35  
Hurford, J., 1, 13–15, 17, 19, 21–25, 27–32, 34, 36, 37, 39, 41, 42
- I-Language, 3, 40, 41  
incremental learning, 30, 31  
induction, 4, 30–32, 35, 41, 42  
instance-based learning, 18  
intuitions, grammatical, 40  
invention, 3, 7, 25–32, 34, 41, 42
- Johansson, C., 36
- Kegl, J., 25  
Kirby, S., 1, 13–15, 17, 19, 21–25, 27–32, 34–37, 39, 41, 42
- Langley, P., 18, 30, 31  
language change (historical), 4  
Lightfoot, D., 3  
linear ordering, 2, 23, 29

- mapping of meanings to forms, 21–23, 32
- meaningless substrings, 22, 29
- meanings, 5
- meanings, representation of, 20–25, 29, 41, 42
- multi-generational E/I models, 36–38, 41
- Nattinger, J.R., 19
- neural net, 38
- neural nets, 4, 20, 24–26, 33
- Niyogi, P., 4, 5
- noise, 5
- noun phrases, 23, 24
- Oliphant, M., 7, 36
- parsing, 20, 30
- passive construction, 23
- Peters, A., 19
- phonetic elements, 6
- population dynamics, 36–38, 42
- population size, 36
- populations of agents, 3
- production (of utterances), 3, 26, 27, 31, 32, 40
- proper names, 23
- protolanguage, 27
- recursion, 2
- Roberts, I., 5
- rote learning, 13
- rules, 16–20, 31, 33, 41
- Senghas, A., 25
- social coordination, 37
- sound-symbolism, 25
- spatial organization, 36
- Steels, L., 7
- synonymy, 9, 10, 23, 34
- syntactic categories, 15–17
- syntax, 11
- syntax, autonomy of, 15
- syntax, definition of, 2, 11–13, 15, 16, 20
- syntax, emergence of, 13, 14, 16, 19, 20, 24, 25, 27, 37, 39, 41, 42
- syntax, evolution of, 1
- syntax, representation of, 14
- synthetic route to complex language, 27
- uni-generational E/I models, 36–38, 41
- VanLancker, D., 19
- variables, 18
- vocabulary, 7–10
- Vogt, 7
- Weinert, R., 20
- Wexler, K., 5
- Widdowson, H., 19
- Wray, A., 20
- Yorio, C., 19