Word Order and Case in Models of Simulated Language Evolution

Joanna Moy

This thesis is submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

University of York
York
YO10 5DD
UK

Department of Computer Science

March 2005

Abstract

An attempt to simulate the emergence of case-like behaviour in populations of communicating software agents is presented. An implementation of an Iterated Learning Model is described based on Kirby [44]. The occasional emergence of grammars with distinguishable subject and object noun categories is noted. Changes are then made to this model to enable the use of multiple word orders, in the hope that this will promote such behaviour.

These changes do not appear to promote the emergence of such grammars to the degree anticipated, due to the fact that there is no requirement for agents to understand each other. Further changes are made to the model such that learners will reject utterances which they believe to mean something other than what the speaker intended. As a result, there is a rise in the relative number of two-noun category grammars emerging, although the changes to the model also have a destabilising effect, resulting in a decrease in the absolute number.

Experiments are also described involving external manipulation of the subset of the meaningspace that agents are permitted to use, also known as the learning bottleneck. The results of this appear to show that in the presence of a very strong bottleneck, regular and fully compositional grammars with a single noun-category are favoured, whilst relaxing the bottleneck to an intermediate value seems to promote the emergence of two-noun category grammars.

Finally, a different approach to the emergence of case is described. This involves attempting to achieve proper inflectional case markings in the absence of free word order by treating inflection as another level of compositionality. The emergence of inflectional endings again seems to be favoured by the imposition of a moderately sized learning bottleneck.

Contents

List of Figures

Acknowledgements

Firstly I would like to thank my supervisor, Dr. Suresh Manandhar, for the instruction and guidance he has given me over the past six years. I would also like to thank Yann Golanski, Richard Clegg and Zoe Stephenson for their advice and help with proofreading this thesis.

Secondly, I owe a great deal of thanks to my parents for the support they have given me, both emotional and financial. Without their help this work would undoubtedly never have been completed. I would also like to thank them for bringing me up to have an enquiring mind and instilling in me a love of learning from an early age.

Thanks are also due to the other members of the Artificial Intelligence Group in York, for their general advice, expertise and encouragement. In particular, my fellow PhD students and good friends Stephen Watkinson, Dan Sheridan, Lyndon Drake and Jose Luis Jara Valencia who made time spent in the Computer Science department so enjoyable. I am also indebted to the various academics from other departments and institutions who have helped along the way, in particular the members of the LEC in Edinburgh.

I am also very thankful to the many, many wonderful friends outside my academic life without whom this experience would not have been as rich as it has been. Your faith in me has been much appreciated, as has your help in getting me through the tough times. To Ed, Chris and Nick, I would like to say thanks for providing me with a most enjoyable diversion in the form of the Screaming Banshee Aircrew, and small taste of goth-rock stardom! To Andy, Mel, Simon and Mike, I would like to say thank you for your impact on my life during the various stages of this thesis: you have each had your part to play in the final outcome. Finally, and most importantly, I would like to express much gratitude to Richard for his continued love and support – I couldn't have done it without you.

In memory of Rachel

Declaration

This thesis has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any other degree than Doctor of Philosophy of the University of York. This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by explicit references.

I hereby give consent for my thesis, if accepted, to be made available for photocopy and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .....(candidate)

Date.....

A full list of papers published by the author pertaining to this thesis is contained within the list of references.

Chapter 1

Introduction

Human language is a communication system quite unlike any other found in the natural world. It has the unique property of allowing us to convey complex propositional statements about things both present and absent, near and distant, past and future, real and imaginary. It is something that sets our species apart from all other animals, by allowing us to communicate not just about things that are relevant to our immediate well being – food, danger, courtship, ownership of territory – but also about anything else that we choose. It is arguably one of the key features that makes us human.

Unsurprisingly then, the origins of this peculiar human skill have engendered a great deal of interest. How did language come about? What is it about the human brain that makes language possible? How is language learnt? These question are significant to Linguists, Psychologists and Computer Scientists alike, and there is a great deal of inter-disciplinary interest. Quite apart from an interest in the problem for its own sake, if Linguists and Psychologists are able to provide a better understanding of the structure of language, the way in which it emerged and the nature of the mechanism by which it is acquired, this knowledge could be potentially very useful in the development

of computer systems that can learn a language or communicate with the human user. Conversely, Computer Science has also proved a useful tool to linguists, by providing methods by which the acquisition of language can be modelled in populations of communicating agents.

Up until now syntacticians have had very few scientific methods at their disposal when it comes to the emergence of language. It is a phenomenon that has evolved once and only once, as far as we know, and there are no species with nascent language faculties for us to observe. The only means available has been to study the acquisition of language in individuals of our own species, in the hope that this might represent some kind of microcosm of the emergence of language in the species as a whole. Also, studying the way in which language is learnt can help to elucidate its fundamental structure, and this in turn can give an insight into the cognitive structures that are necessary for the acquisition of language, which might suggest ways in which it evolved. However, this avenue too, is somewhat limited in the tools it can use – certainly experimentation on human children to see what is necessary for the successful acquisition of language in terms of linguistic input and other stimuli is not an option. Observing the way in which children learn language can be helpful, but leaves little room for *testing* hypotheses. As a result, research has often been limited to simply theorising about what the key features of any language-acquiring mechanism may be, and attempting to see if these theories are compatible with the utterances of real human children at various stages of learning their first language.

However, more recently, the use of computer models has been able to add to the body of knowledge provided by such psycholinguistic investigations, and has given some valuable insights into the mechanisms by which language may have emerged and the way in which it is acquired. This is the subject matter with which this thesis is primarily concerned.

One of the fundamental questions regarding the acquisition and evolution

of language, is the degree to which it is innate. Whilst it cannot realistically be argued that humans beings are not in some respects predisposed to language, the debate still rages with regard to exactly how much and what aspects of our language capacity is genetically endowed. The traditional Chomskyan view [19, 23] is that we have a highly specialized language faculty that determines the precise nature of language. Acquisition is simply a matter of selecting the correct parameter settings for the local language from those made available by this language faculty. This notion is drawn from the observation that although languages differ significantly, the commonalities between them are striking, and only a very small proportion of the space of logical possibilities seems to be covered. Chomsky's second supporting observation is the claim that the nature of language data available to children radically under-specifies the precise nature of language (although some would argue that this is not in fact the case e.g. [64]). If the structure of language is not made explicit in the learning environment, and yet human children are able to acquire it with such accuracy and ease, then knowledge about how languages are structured must surely be innate?

An alternative viewpoint that has found favour in more recent years is that whilst we obviously do have *pre-adaptations* to language which give us both the desire to communicate verbally, and the aptitude to do so, we do not have any *specific* innate knowledge about what language is like. The fact that children are able to acquire language despite the under-specified nature of the data is due to the fact that languages themselves are evolving entities that have been shaped by the biases of human learners. Thus whilst Chomsky observes that children appear to make "lucky guesses" during the acquisition process as to what language should be like, and attributes this to innate knowledge of its structure, Deacon [27] suggests that these guesses only *appear* to be lucky because languages have evolved such that the guesses children are likely to make will most probably be correct.

This is one arena in which the use of computer models mentioned above

has really come into its own. Several authors [46, 3, 70, 85] have succeeded in demonstrating the emergence of compositional syntax in populations of language learning agents with no innate language-specific knowledge. This is attributed to the learning biases of the agents themselves, as described above, and also to the *dynamics of language transmission*, in particular what is known as the "language bottleneck": under circumstances where agents cannot hope to hear all possible utterances from a language during their lifetime, languages in which the meaning of the utterance can be predicted from the meaning of its parts will have a selective advantage. Further studies along these lines have succeeded in demonstrating the emergence of other characteristics of natural language also, such as recursion [44], patterns of stable irregularity [47] etc. The aim of the current investigation is to ascertain whether it is possible to add more complex aspects of generative syntax to this repertoire, namely the use of case to signal semantic relationships.

Most languages of the world employ one of two mechanisms to signal semantic relationships (that is the details of "who did what to whom") between participants in events or actions being described: either word order, as in English, or case-markings as in languages such as Latin and Russian. Sometimes other cues are brought to bear such as the use of pragmatic cues and verb-subject agreement, but word order and case are the two primary methods. Case-markings are typically manifested as either inflectional affixes attached to noun phrases (usually just the head noun, but sometimes also articles and adjectives) or as grammatical function words denoting role, such as the use of prepositions in English to specify direct and indirect objects of ditransitive verbs. In the computational studies of language evolution described above, it is notable that semantic relationships are universally signalled using word order. None of the languages emerging exhibit any of the properties of case grammar. This is clearly a deficit as it does not reflect the true nature of human language. Thus if we are to be able to take seriously this account regarding the dynamics of language transmission and the biases of the learner,

we must show that case-grammars are an equally plausible outcome of simulations such as these. This is the aim of this thesis.

In the following chapter I shall describe in more detail some of the current theories of language evolution, before going on in Chapter 3 to review the history of modelling the acquisition and emergence of language using computer simulations, and in particular, a type of model known as the "Iterated Learning Model" [9]. In Chapter 4, I shall describe the details of my own implementation of such a model and the results of this. In Chapter 5, I will develop this further, and by introducing a degree of word order freedom to the utterances produced by agents in the simulation, I will attempt to promote the emergence of a primitive kind of case system. In Chapter 6, the shortcomings of the results of this will be discussed, and further changes will be made in order to overcome these. Finally, in Chapter 7, a somewhat different approach will be described in which the evolution of a proper system of inflectional marking is attempted. The results of the experimental work described in these chapters will be drawn together in Chapter 8.

Chapter 2

Theories of Language Evolution

2.1 Nature or Nurture?

That human beings are in some respect predisposed to learn language is beyond any reasonable doubt. In his fascinating book, The Symbolic Species [27], Terrence Deacon describes how our species manifests an extensive array of perceptual, motor, learning, and even emotional predispositions towards the learning of language. For example, the human larynx is positioned significantly lower in the throat than that of other primates, which puts us at much greater risk of choking on our food, but increases the range of sounds that can be produced by allowing greater changes in the volume of the resonant chamber, and also by shifting sound away from the nose and towards the mouth. Furthermore, we have a much greater degree of voluntary control over our respiratory function than other primates, enabling the long slow exhalations necessary for the production of speech. And although the neural correlates of a predisposition to language are less easy to identify than the physiological ones, it is worth noting that all normal children raised in normal social environments inevitably learn their local language, whereas other

species, even when raised and taught in this same environment, do not [27]. Language emerges in all normal children all over the world at approximately the same age, no matter what culture they are growing up in and what language they are acquiring [2]. Furthermore, Briscoe [15] notes that "failure [to acquire language] appears to correlate more with genetic defects or with an almost complete lack of linguistic input during the critical period, than with measures of general intelligence or the quality or informativeness of the learning environment" (p. 1). There are certainly many examples of individuals, whether through lack of input or due to brain damage, have extremely impaired language capabilities despite normal levels of intelligence [25, 51], and conversely severely brain-damaged individuals who are extremely proficient with language [92]. As mentioned previously, it would appear that human language is quite unique: the study of communication systems in other animals has failed to identify anything comparable (see Aitchison [2] for an overview), and attempts to teach language to non-human primates, whilst showing that they may be capable of mastering symbolic representation to a limited degree, demonstrate that they are certainly not capable of human proficiency, and that they certainly do not appear to have the same "natural aptitude" that we do [66, 39].

It is clear that there is "something special about human brains that enables us to do with ease what no other species can do even minimally without intense effort and remarkably insightful training" [27]. So perhaps the real question to be addressed is not whether language is the product of nature or nurture, but which *aspects* of language *are* innate? What form do they take, and how did they arise?

2.2 Universal Grammar

One of the most interesting observations about human language is that despite the vast array of different languages being spoken around the world, they are all essentially remarkably similar. Even those which appear incredibly variable on the surface seem to share a vast array of commonalities when inspected in more detail. Noam Chomsky argues that this is because all human languages are underpinned by a Universal Grammar (UG), knowledge of which is in some sense part of the innate endowment of a child [19]. In "Aspects of the Theory of Syntax" he argues that the primary linguistic data available to children is not sufficient to make explicit the underlying structure of the language, and that learning is therefore guided by a specific cognitive module, which he calls the "Language Acquisition Device" (or LAD). Universal Grammar is at the very core of this specialised module, providing detailed information about the space of possible languages. The problem of acquiring a language is thus reduced to finding the correct grammar from within this space.

He claims that the existence of UG and the LAD are necessary to be able to explain the phenomenon of language acquisition. The main thrust of his argument is that language acquisition is not simply a matter of learning sentences by rote and repeating them, but that it requires the internalisation of a set of rules for the construction of novel utterances. The underlying structure of language is very complex, and it is not made explicit in the language data available to children [20], thus rendering the acquisition process intractable without extensive trial and error learning with explicit feedback. There are many studies that suggest that children are not reliably provided with negative feedback on their incorrect utterances [17] and that even when they are, they are unable to make use of the information [8], and yet young children rapidly and easily learn the complex rules of grammar. Theoretical studies such as Gold [35] have shown that without negative feedback, infi-

nite languages are effectively unlearnable without constraints on the set of targets. Thus Chomsky believes that children *must* be endowed with innate knowledge about what languages must be like.

In his later work [22], he develops the "Principles and Parameters" model as an alternative to UG, in which grammatical constructs can be divided into principles, i.e. those which are invariant across languages (such as the existence of syntactic categories such as nouns and verbs), and parameters, of which there are a finite number, each with a finite number of values (for example, whether adpositions preceed or follow a noun phrase). Thus, the acquisition task is further simplified to finding the correct settings for each of these parameters.

The observation that the problem of learning language is a much more difficult one than the experience of countless generations of human children would suggest has also been made by other authors. William O'Grady [57] presents a discussion of the relationship between the experiences of language available to children and the nature of the grammar they extract from it, suggesting that the experience to which children are exposed radically under-determines the type of grammar required to be able to speak and understand a human language; this is an example of the "projection problem" whereby the task of the learner is to try and determine underlying regularities in the language to which he/she is exposed (i.e to acquire a grammar) when the data available may not be sufficient to determine these uniquely. He is able to identify a number of areas of syntactic structure where this is the case, including the syntactic categories to which words belong and their properties, the underlying hierarchical structure from which sentences are composed, and the constraints on form and interpretation of sentences such as placement of gaps and resolution of pronouns.

Clearly it is easy to suggest that if the structure of language is underdetermined by the experience of the learner and that some sort of innate

knowledge must be making up the shortfall.

2.3 Natural Selection of the Language Faculty

But what form does this innate knowledge take? Is it highly structured linguistic information as Chomsky suggests? Or is it simply the product of more general human cognitive capacities? One argument against the notion of a Universal Grammar and a Language Acquisition Device is the question of how such a thing could have arisen. Many theorists believe that it is far too complex to have emerged by Darwinian natural selection, and would have required some kind of "catastrophic mutation" to bring it into existence. Chomsky himself is rather vague on this subject, not wishing to attribute it to natural selection but instead to complex processes, as yet not fully understood, and "reasons that have to do with the biology of cells, to be explained in terms of properties of physical mechanisms" [23] (p. 169). The essence of the argument that Universal Grammar could not have arisen by natural selection is that a "partial" grammar faculty would be of little value to an organism.

Pinker, however, argues that the existence of language can *only* be explained as a result of Darwinian natural selection. Although he does not support Chomsky's belief in a very highly determined Universal Grammar, he does believe that we are endowed with a complex language faculty that provides us with a large amount of innate information about the nature and structure of language, and argues that this must have arisen by natural selection because it is the only process that can steer organisms through a myriad of possibilities to give the appearance of design that is so apparent in all living things. The alternatives can only grope randomly he claims, and he likens trying to attribute such complexity to the proverbial hurricane that blows through a junkyard and assembles a Boeing 747 [62]. In his much-cited paper

of 1990, co-authored with Paul Bloom, he says:

"Evolutionary theory offers clear criteria for when a trait should be attributed to natural selection: complex design for some function, and the absence of alternative processes capable of explaining such complexity. Human language meets this criterion..."

Steven Pinker and Paul Bloom, "Natural Language and Natural Selection" [63] (p. 1)

He presents a series of comprehensive criticisms of those authors who have suggested that language may have developed by other mechanisms, using the premise that language is a feature of our biology, not society and that natural selection is the only successful mechanism by which biological complexity can be accounted for. He draws analogies between the development of language and the evolution of the eye, rejecting the idea that the language acquisition device is merely a by-product of selection for other cognitive abilities. He also rejects Gould's notion that language is a "spandrel" [37] or Bickerton's view that it could not have evolved unless by a single "catastrophic mutation" [5], due a lack of selective pressure for the partial ability to acquire language. In answer to this, Pinker and Bloom quote Gould [36] as having said "What good is 5 per cent of an eye?" and Dawkins' [26] reply: "An ancient animal with 5 per cent of an eye ... used it for 5 per cent vision. ... Vision that is 5 per cent as good as yours or mine is very much worth having in comparison with no vision at all. So is 1 per cent vision better than total blindness. And 6 per cent vision is better than 5, 7 per cent better than 6, and so on up the gradual, continuous series". So it is also the case for language evolution, they claim.

2.4 Alternatives to Universal Grammar

However, if the highly specific nature of Universal Grammar really does pose a problem for arguments suggesting that the language faculty evolved by Darwinian Natural Selection, how might we get round it? There are a number of writers who have tried to suggest alternative forms that an innate language faculty might take and ways in which might have evolved. Some of their arguments will be reviewed below.

2.4.1 General nativism

William O'Grady [57, 58] takes the position commonly known as "General Nativism". That is, he rejects the idea of a Universal Grammar, but also the completely inductivist view, that the structure of language is elicited purely from the experience of the child. Instead he proposes the existence of a "General Nativist Acquisition Device", made up of a number of modules which interact with each other to help the child elucidate grammar for the utterances heard in its linguistic environment. He claims that some of these modules may be specific to the language faculty whereas others may also have independent non-linguistic functions, and might have evolved to fulfil quite different purposes.

Within this framework, he aims to provide an alternative solution the learnability problems previously discussed in relation to Universal Grammar. He stresses that any truly plausible alternative must include a system of sentence formation that accounts for the full range of syntactic phenomena found in adult speech, and an explanation as to how these phenomena might arise without the need for the highly determined innate linguistic structure that Chomsky proposes. In particular, he aims to find a plausible solution to the problem of acquisition of those features of language which he believes are

not made explicit in the utterances children hear, as previously mentioned in Section 2.2: syntactic categories, hierarchical structure and constraints regulating phenomena such as gap placement and reflexive pronouns.

The General Nativist Acquisition Device hypothesised by O'Grady is based on five modules. The most fundamental of these are the *perceptual* module, responsible for the analysis on the incoming auditory signal, and the *learning* module, which provides mechanisms by which hypotheses on the structure of language can be formed and tested. On top of these, he first proposes a *conceptual* module for the acquisition of syntactic categories which identifies members of each category from their semantic correlates. This idea is similar in some ways to Pinker's theory of Semantic Bootstrapping [60]. Unlike O'Grady, Pinker believes children have innate knowledge of the existence of syntactic categories such as nouns and verbs, but the problem they are faced with is identifying *which words* are examples of which category. Whilst it is generally accepted that grammatical concepts such as these do not have reliable semantic identities, it is suggested that perhaps this is not the case in parent-child discourse; that when speaking to children, adults use nouns only to refer to people and objects, and verbs to refer only to actions and changes of state etc. Thus in the early stages of language acquisition, the child is able to identify syntactic categories by these semantic properties. Once the distributional properties of the categories have been learnt, it will then be possible to widen their semantic scope to include the full range seen in adult language.

In a similar way, O'Grady proposes that syntactic categories can be reduced to their underlying semantic notions. However, he suggests that the categorisations adopted by the Semantic Bootstrapping hypothesis are unnecessarily restrictive. Rather than allowing only *some* instances of syntactic categories to have semantic correlates, he instead proposes a set of broader categories: verbs as *events* (encompassing both actions and changes of state), nouns as *individuatable things* (which includes verbal nouns such as "a walk") and

adjectives as *gradable properties*. This, he claims, leads all instances of a particular category to have the same properties. Therefore, rather than containing innate knowledge of the existence of verbs and nouns, the language module responsible for syntactic categories instead has information about each of these three notions, as well as information about the distributional properties of each: *events* are grounded in time and therefore likely to be associated with time-based markings such as tense and aspect; *individuatable things* are likely to be associated with determiners or deictics (pronouns such as this and that) responsible for individuating them; *gradable properties* are associated with those morphemes that a language uses to indicate gradation ("too", "very", etc. in English). Armed with this information, this cognitive module is able to classify the terms it encounters into categories.

The fourth cognitive module that O'Grady proposes is a *propositional* module responsible for representing meanings in a manner similar to Fodor's "language of thought" [33]. This helps elucidate the hierarchical structure of language by providing information about predicate-argument relations, the nature of those arguments (agent, theme, goal), and distinctions between past and non-past, definite and indefinite, etc. He claims that this representation is not specifically syntactic – it contains no syntactic labels or phrasal constituents. By combining information on number and types of arguments required by different predicates with the results of the classifications performed above by the perceptual module (*event* vs *individuatable thing* vs *gradable property*), a simple lexicon can be built up. Each entry in the lexicon appears either as a basic category i.e. one that does not require any arguments, or as a functor – one with dependencies (or unsatisfied arguments).

Functor categories then combine with argument categories of the appropriate number and type, as regulated by the final *computational* module. O'Grady proposes this module as an alternative to the classical Government and Binding Theory (GB) [22], saying that the combinatorial operations are driven

by semantic considerations rather than syntactic ones. He suggests that this module has three important properties (which are also commonly manifested in other syntactic frameworks such as GB): binarity (application of the combinatorial operation to pairs of elements), inheritability (dependencies which are not satisfied by the operation in question are passed up to the next level) and iterativity (the same operation can be continuously re-applied until all dependencies are satisfied).

Again, O'Grady argues that this computational module may not be language specific, but instead is the same cognitive structure that we use to perform arithmetic operations, substantiating this claim with the observation that the our "arithmetic faculty" also displays the properties such as binarity.

Finally, the question of how the learner is able to discover the principles underlying issues such as gap placement and pronoun resolution is raised, which O'Grady claims is one of the biggest challenges for any general nativist theory of language acquisition. However, he attempts to make a start on this with a proposition on how reflexive pronouns might be dealt with. For example in a sentence such as "Harry$_i$ admires himself$_i$", *Harry* and *himself* co-refer. However, in the sentence "Harry's brother$_i$ admires himself$_i$", *himself* refers not to *Harry* but to his brother. Such a sentence where *Harry* were the intended co-referent would be ungrammatical in English, and indeed all languages, according to O'Grady. This poses the question of how such a principle is learnt without innate syntactic information of the type hypothesised in UG. O'Grady proposes that in addition to the dependencies between functors and arguments discussed above, there is a further type of dependency which must also be satisfied when lexical items combine to form sentences. These he calls "interpretative dependencies", whereby one element must look to another for determination of its reference. Elements may possess one of two types of referential index: a functor index, which occurs on a reflexive pronoun, and must co-refer to an index of the other type, the basic index, which occurs on lexical noun phrases.

Thus the path of an interpretative dependency can be traced as a sentence such as "Harry$_i$ admires himself$_i$" is formed. Firstly *admires* combines with *himself* to provide the transitive verb with a theme argument. At this point, the functor index on *himself* does not refer to the basic index on another noun phrase, and is as yet unsatisfied, causing it to be inherited by the resulting verb phrase *admires himself*. This phrase also inherits the unsatisfied requirement for an agent argument for the verb *admire*. When the whole phrase *admires himself* is combined with *Harry*, both dependencies are satisfied. In the case of sentence such as "Harry's$_i$ brother$_j$ admires himself$_j$", the verb phrase *admires himself* would be combining with the whole noun phrase *Harry's brother*, which carries the basic index not of *Harry* but his brother.

The situation regarding the binding of reflexives in relative clauses is more tricky however. Take for example a sentence such as "Peter$_j$ says that John$_i$ admires himself$_i$". In English, the reflexive pronoun *may only* refer to John, not to Peter. However, in Japanese, it could refer to either. In GB, this is generally accounted for the by the existence of a grammatical parameter which specifies whether the reflexive pronoun must be bound within the relative clause, or whether the referent can be outside of it. O'Grady proposes that the *learning* module of his acquistion device is equipped with a law of conservatism, requiring it to adopt the most conservative hypothesis consistent with experience. Thus, in English, this module would assume that co-referents for reflexive pronouns must be within the same clause as the pronoun itself, and would never receive any evidence to the contrary, whilst in Japanese, it would soon be alerted to the fact that the most conservative hypothesis was incorrect.

Thus O'Grady concludes by stressing that his proposal for a General Nativist Acquisition Device does not require the categorical and hierarchical properties of language to be somehow discovered by experience as a purely empiricist theory would do, but instead follows inborn principles and prop-

erties. However, the contrast with approaches such as Universal Grammar is that these principles and properties are not specifically syntactic in character, and may be manifested in other aspects of cognition.

2.4.2 Proto-language

Yet another alternative to UG is that of Derek Bickerton [5], who proposes a two-stage hypothesis for the evolution of language: a proto-language stage, which was probably fairly stable and long-lived, followed by the emergence of fully developed modern human language. This, he claims, happened quite rapidly [6], through the use of pre-existing brain structures that might have evolved individually for other purposes, but that collectively could have been considered a pre-adaption for syntax. This is perhaps is an idea not too dissimilar from that of O'Grady's General Nativist Acquisition Device [57, 58], just discussed. For example, he postulates a brain module for "thematic analysis" in terms of keeping track with "who did what to whom" which could be very useful in a highly social and intelligent species such as ours, even prior to language, but that could very well be put to good use in the assignment of semantic roles (c.f. O'Grady's "propositional module"). He believes that the proto-language stage was composed largely of stand-alone lexical items of a referential nature – names for things and simple actions, which were combined without syntax to produce primitive utterances. As evidence for this, he claims that remnants of the proto-language system can still be seen today in situations where full language has not (yet) developed or is disrupted, for example, the one- and two-word utterances of pre-linguistic children, the symbolic language of chimpanzees, and pidgin languages. Furthermore, he claims that such a proto-language is still part of the human psychological endowment, remaining accessible to all members of the species throughout life, which explains the rudimentary language skills that children such as the unfortunate "Genie" are able to master, who had been isolated from human

contact until her teenage years – well past the critical period for normal language acquisition [25]. He also believes that modern humans also tend to revert to this mode of communication when we are feeling cognitively incapacitated as a result of fatigue, alcohol, sickness etc.

Alison Wray [88], although another strong proponent of the theory that full modern syntax arose from a primitive proto-language, argues that there are many problems with this view. First and foremost she takes issue with the existence of single referential words, saying it is not at all clear where these words would have come from. Furthermore, she suggests that it is difficult to imagine what advantage such a primitive half-way grammar would have for its users compared to the highly successful communication systems used by all modern primates, and therefore presumably by our hominid ancestors. Instead she proposes the existence of a *holistic* proto-language. She suggests that early hominids living in large social groups might well have used holistic utterances to communicate about the kind of thing that modern-day chimpanzees do, but that perhaps they had a need for a more complex inventory of functional exchanges. Utterances would primarily have been used for building and maintaining social relationships, in a way that is reminiscent of Dunbar's theory of social grooming [30]. Given that these utterances are holistic, each would have to be auditorily distinct. Thus this would create the need for ever more phonemically complex sounds. This would have two effects: firstly to drive the evolution of the human vocal tract to be able to produce the wide range of different phonemes that we are capable of today (which is arguably more than we actually *need* for the production of modern language – hence the fact that phonemes exist in some languages that are absent in others); and secondly to possibly result in longer, polysyllabic utterances. This, she suggests, might act as a foothold for the evolution of full modern syntax.

Thus we have the basis of a useful and stable communication system that may well have persisted for several million years. What next? How did full

modern syntax develop? Like Bickerton, Wray seems to advocate the point of view that the ability to process syntax may have arisen through the linking together of specific brain structures that had originally evolved for other purposes. Unlike Bickerton, she does not believe that the change from proto-language to modern language was a sudden one. She posits instead that this was a gradual process which arose due to segmentation of holistic utterances. She gives the following example as to how this could occur: supposing that in the proto-language, the utterances /mɛbita/ and /ikatubɛ/ mean *give her the food* and *give me the food*, respectively, and also that the utterance /kamɛti/ means *give her the stone*. Of course, these three utterances are completely arbitrary; any similarity between the syllables used is purely coincidental. However, by chance /mɛbita/ and /kamɛti/ share the syllable /mɛ/ and also share in their meaning a single female recipient. Thus an early human proto-language user, perhaps driven by Bickerton's hypothesised theta-role finding brain module, might arrive at the conclusion that this syllable means *her*. Furthermore, the utterances /ikatubɛ/ and /kamɛti/ both share the syllable /ka/ and refer to the act of giving; thus the same individual may well be drawn to postulate that /ka/ means *give*. Of course, just because one or two language users have noted that there are some similarities between arbitrary strings with related meanings and chosen to segment their strings in this way, that doesn't mean we're at the full compositional language stage just yet. For a start, for every coincidental similarity between utterances, there will potentially be many counter-examples which suggest that the segmentation being hypothesised is incorrect, such as other utterances involving female recipients that do not contain the syllable /mɛ/. Wray suggests that there are several possible outcomes that may result from this: a) the individual may conclude that they were wrong and that /mɛ/ does not mean *her* after all; b) that overgeneralisations will occur – perhaps leading to the changing in pronunciation of a similar syllable in another utterance so that it starts to *sound like* /mɛ/; or c) there will be overgeneralisation in the opposite direction, for example in the case of an utterance that appears to refer to the act of giving

but does not contain the syllable /ka/, which might lead the individual to make subtle semantic distinctions, such as concluding that the utterance in question may refer to the act of *presenting* rather than merely *giving*. Thus, slowly but surely, over the course of many generations, differences in use and pronunciations of utterances will creep into the language (just as they do in modern languages) which reinforce and aid the compositional interpretation of strings, and ultimately a fully rule based compositional language will result.

Another viewpoint that Wray has in common with Bickerton is the belief that proto-language has never been fully be replaced, but still exists in modern-day speech. However, once again, she disagrees with him over the details: she dismisses his examples of proto-language, such as the one- and two-word utterances of young children, the speech of adults deprived of linguistic input during childhood, pidgin languages etc. because they are all examples of one "proto-language" user interacting with fully competent adults. Instead, Wray proposes that the remnants of proto-language can be seen in the everyday formulaic speech that we all use very frequently – holophrases such as "How do you do?", "Can I help you?", "Give it a rest!" etc. Whilst it is undeniable that these phrases share the structure of modern syntactic language, they are not used in this way – they are not generated from rules each time they are used, and have meanings which in the social context are far wider than the *literal* meaning of the phrases themselves. Wray has written extensively on this topic e.g. [91, 90].

2.4.3 Incremental language evolution

Another variation on the idea of proto-language is Ray Jackendoff's theory of language evolution [42]. He takes a similar viewpoint to Pinker, suggesting that the language faculty could *indeed* have arisen incrementally

through selective pressure for communicative ability. He urges us to consider the Chomskyan Language Acquisition Device not as a single, all or nothing "grammar box", but rather a "toolkit" for language acquisition, composed of many discrete capabilities. Thus he argues, it is perfectly possible to view it as something that might have arisen through natural selection. He proposes a two-tier theory of language evolution not dissimilar to, but more elaborate than, the proto-language theories outlined above.

According to his hypothesis, the main pre-requisite for proto-language is a population of individuals with thoughts worth communicating to each other. He believes that this is something that can already be said of chimpanzees, on account of their rich social structure and problem solving abilities, and thus it is easy to imagine that it may also have been true of our last common ancestor with them. He goes on to suggest that the most important step for the emergence of language in such a population is the use of symbols; however, these symbols must be used in a non-situation specific manner, unlike some of the referential symbols used by contemporary non-human primates, such as the vervet alarm calls. It might be argued that some great apes can be trained in this ability [81, 66], and thus that they may possess some of the cognitive pre-requisites for language. Once this ability has arisen, he claims that there are two important innovations that need to occur to set a species on the path to language – firstly the ability to use a open, unlimited class of symbols (and the appropriate phonological support for this), and secondly, the concatenation of symbols, which could be viewed as the beginnings of syntax. These two changes are logically independent, and could have arisen quite separately.

Unlike many writers, he believes that each of these preliminary stages in the putative evolution of language are adaptive in their own right, and that they can each confer a fitness advantage in terms of improved communication. He compares them to the "one word" stage of a baby's development; at this point, utterances may sometimes be difficult to interpret, but nonethe-

less, communication *is* occurring, and is a lot more productive than before the child has any linguistic competence at all. Once the two-word stage commences, i.e. symbol concatenation has begun, communicative ability is enhanced further still. Thus, he believes that such behaviours in primitive humans could each have been suitable targets for natural selection, and there is no need to suggest that they must have arisen simultaneously or as one large package containing the whole language faculty.

Finally, in order to have a workable "proto-language" of the type that Bickerton describes, Jackendoff claims that the primitive use of word order to express basic semantic relations is necessary. In the early stages of symbol concatenation, it would probably be fairly trivial to figure out from the context exactly what the intended sense is. However, as more words are added, the number of pragmatic possibilities starts to increase, and even with the knowledge of the context it may not be possible to precisely determine the intended meaning. Nonetheless, as Jackendoff rightly points out, it is not necessary to have full generative syntax in order to improve this situation. He suggests that modern languages display some robust principles of word ordering which are in some sense outside of their syntax, and can be seen quite clearly in situations when normal syntax is somehow impaired. He discusses the early stages of first language acquisition, pidgin languages, and the impoverished speech which Klein and Perdue refer to as "The Basic Variety" [50] that is seen in adult second language learners who have received no explicit instruction in their target language. In all these cases, there seems to be a tendency to exhibit an "agent-first, topic-last" type behaviour. There is a suggestion too, that some of the "talking chimpanzees" used in ape-language studies also follow this rule [67]. Thus Jackendoff suggests that it is a "fossil principle" incorporated into the Universal Grammar toolkit during the proto-language stage.

Thus we have a plausible story starting to emerge; a set of pre-requisites for language, some of which currently exist in other non-human primates, and

thus could easily have also occurred in our ape-like ancestors, and a set of possible small innovations that would have led to the incremental development of proto-language. Why is it then that our ancestors did indeed develop such a proto-language when other apes did not? If chimpanzees possess some of these pre-requisites in the form of thoughts worthy of communication and the ability to use symbols, and possibly even the tendency to use basic word order to express semantic relations, why do they not speak also? Presumably due to the absence of selective pressure in their evolutionary environment for the crucial innovations described above. To the extent which they do display some of the cognitive structures that may underlie language, it is quite clear that they have nowhere near the proficiency that we do, even in the most basic skills.

Let us look at the ability to acquire a large open class of symbols. Although some primates can be taught to use symbols in a non-situation specific manner, to a limited degree, they certainly lack the human aptitude for it, as exemplified by the huge difference in the relative sizes of the average human vocabulary and that of chimpanzees and bonobos who have been used as subjects in ape language studies [81, 66, 67]. If, at some point in our evolutionary history, there was a good adaptive reason for us to be able to use a very large number of symbols, this would have resulted in a selective pressure for the ease of vocabulary acquisition that we see in modern human children – something they do rapidly and with minimal instruction, quite unlike the majority of apes who have been trained in symbol use. Human behaviours which may underlie this skill include the ability to imitate and the use of pointing, skills which are not present in the great apes. Thus, it might be possible to consider these skills as part of the Jackendoff's Universal Grammar "toolkit", the set of cognitive pre-adaptations necessary to kick-start the evolution of language.

So, at this stage in the hypothesis, we have a population of individuals communicating with each other in "proto-language", putting together meaning-

ful symbols to form larger utterances whose meanings are a function of the meanings of the constituent symbols. Already, selective pressure for communicative ability may have resulted in a number of aspects of the Universal Grammar being added to the toolkit, including the tools necessary to be able to quickly and easily learn symbols and the ability to invent new symbols for new situations resulting in an open vocabulary, the concatenation of symbols to express more complex propositions whose meanings are a function of the meanings of their parts, and the use of word order and to express semantic distinctions in these complex meanings. What happened next? How did full modern language evolve?

Unlike Bickerton, Jackendoff does not suggest the mystical one-stage "giant leap" from proto-language to modern language, nor does he propose a prolonged single process like Wray, but instead describes of a set of putative stages that the proto-language might have progressed through in order to develop the full generative syntax that we have today. These include the development of phrase structure, syntactic categories (including the noun-verb distinction) and the use of morphological markings. He suggests that each of these may have evolved independently and by quite separate mechanisms.

Throughout the discussion of this hypothesis, Jackendoff stresses two important points: firstly that we need not necessarily appeal to language-specific cognitive structures to explain *all* elements of UG. If there are linguistic universals that can be explained on more general cognitive grounds then we need not ascribe these characteristics to the UG toolkit. And if UG *is* a "toolkit" then it is not necessary to use every tool at ones disposal for every task. To quote: "beyond the bare minimum of concatenated words ... languages can pick and choose which tools they use, and how extensively". For example, modern languages have a wide range of means available to them for the expression of semantic relations – word order, inflectional case markings, and to some degree the use of subject-verb agreement as occurs in some languages. Different languages make use of these particular tools to different degrees,

but not all languages use all of these devices, and some use combinations of them.

2.5 Evolution of Language Rather Than Learner

Returning to Chomsky for a moment: he (and Pinker, to some degree) argue(s) that language must be innate because of complex nature of the data and the ease with which children acquire it when it is so under-specified, claiming that this would only be possible if they have brains that have evolved to provide a large amount of information about the structure of language. However, there is an alternative view point that turns this argument on its head somewhat.

Terrence Deacon [27] suggests that rather than human beings having evolved the necessary cognitive structures to be able to acquire language more easily, perhaps it is the case that *language* has itself evolved to be *more easily acquired* by the cognitive structures present in human brains. He uses a computer-based analogy to illustrate his point – the Apple Macintosh. Up until the advent of this machine in 1984, he claims, the use of computers required extensive training and experience, but the designers of the Macintosh decided to take a different approach – to make interacting with a computer intuitive; to make it possible to learn to use it by trial and error. Since then, all other computer manufacturers have had to follow suit. Users are able to master the basics of a windows based operating system, not because they have innate knowledge enabling them to do so, but because computers have evolved to be compatible with the way people think.

So it is for language, Deacon argues. When, as Chomsky observed, children

quickly master the bases of grammar despite the paucity of the input stimuli from which they are learning, one could perceive them to be making "lucky guesses" about grammar and syntax and the way words work together, which might be all too easy to attribute to some innate, language specific knowledge. Deacon believes that learning is occurring by trial and error, but that a very large proportion of the guesses made are correct – not because of innate knowledge, but because the languages they are learning have evolved in such a way as to fit in with the guesses that children are likely to make.

He says that language can be thought of as a symbiotic relationship between an independent organism and its host, albeit that language is not a truly independent organism, as it lacks its own metabolic processes or the ability to reproduce; however, it can be thought of as more akin to a virus, which is essentially just a package of information encoded as DNA and which relies upon having the information it encapsulates integrated into the machinery of a host cell for its metabolism and reproduction to proceed. In a similar way, the information encapsulated in the grammar of a language becomes integrated into the machinery of the human brain, which reproduces its parts. This relationship is symbiotic (unlike that between virus and host) because both host and language need each other for survival. The metaphor can be taken further still by the observation that although a common language may link a social group, the *internal* grammars of each of those speakers is unlikely to be identical, but subject to variation – it is more like a collection of similar but not identical languages. And as a result of this variation, languages are able to evolve with respect to the selection pressures around them, just as variation within a viral gene-pool drives evolution.

The selection pressures in question here are the biases of human learners. In making their "lucky guesses" about the way in which words work together, children actually neglect a large proportion of the hypothesis space – they fail to explore the full range of ways of organising words. Thus a language that organises its words in a way that falls outside the "lucky guesses" of children

will not easily be learnt and will fail to be passed on to the next generation, whereas a language for whom the "lucky guesses" are usually correct guesses will be much more easily acquired.

2.5.1 Why do languages have so much in common?

So, we have an explanation for the fact that children are able to learn language so easily even though its structure is heavily under-specified by the data. Can this theory also shed some light on the other fundamental observation that led to the idea of a Universal Grammar in the first place: Language Universals?

Deacon believes it can. He suggests that Language Universals are merely an example of convergent features that is a common phenomenon in biological evolution, for example the dorsal fins of sharks and dolphins: structural similarities in unrelated species brought about by the existence of common selective pressures. As a linguistic example, he discusses the common set of colour terms that can be found in many of the world's languages. Not only do most languages possess terms for the same colours of the spectrum, but amongst those that possess fewer, it is always for the same colours. For example, if a language has only two colour terms, these are always light and dark (or white and black). If there is a third, it is always red, and the fourth is always green. Languages with more than four terms tend to add yellow and blue next, brown after that, followed by orange, purple, pink and grey. Furthermore, although the boundaries of these colour terms may vary, (for example, in a language with only 4 terms, blue things might be labelled green), the archetypal example of each is always the same. Deacon asks why this should be the case when we are capable of seeing the whole range of the visual spectrum, capable of inventing a myriad of different terms, and segmenting the spectrum at any arbitrary point?

It might be easy to suggest that this is due to some innate mechanism specifying that these are the important colours that we should know about. However, Deacon claims it is simply due to the biases exerted by the way our nervous system processes colour: because of the response properties of different neurons in the visual processing areas, some colours are more salient to us than others. Thus it is natural that whichever colour distinctions appear first in a language, they should relate to these more salient colours. Moreover, he suggests that if a colour term were to be invented for a shade that was not one of these salient frequencies, its meaning would be likely to slowly change, due to the biases of our perceptual system, until it did describe such a wavelength. Thus, Deacon claims, "the biases our brains introduce...are the social and psychological analogues to mutation". Relatively speaking, he says, the biases introduced by the perceptual system with respect to colour names are quite weak; if they are able to produce such striking similarities between languages, then stronger biases such as the limitations of working memory, attention or sound production should be more than capable of accounting for the observed universals amongst the world's different languages.

Jackendoff [42] is somewhat critical of Deacon's viewpoint, claiming that Deacon puts the cart before the horse. However, it seems that their two points of view might easily be reconciled, especially given Jackendoff's insistence on not viewing Universal Grammar as a single entity, but more a collection of tools, and also that we need not necessarily ascribe *all* features of human language to language specific processes. Similarly, Deacon is quite prepared to accept the possibility of co-evolution of language and learner, and does not deny that human beings may well exhibit language specific adaptations that make the languages which have evolved easier to learn.

2.5.2 A critical period for language acquisition

Another argument that is often used in favour of an innate Universal Grammar is the notion of a *critical period* for language learning.

Certainly, it is demonstrably true that such a thing exists: children seem to acquire languages with relative ease compared to adults, and often show superior ability. Acquisition of a second language seems to occur more rapidly and completely in children than in adults. Similarly, individuals deprived of linguistic input at a young age have difficulty acquiring their first language, and often never achieve full language competency. Interestingly, language learning appears to occur at an age when the learning ability of children is otherwise poorly developed. These observations have lead to the postulation that this "time-dependant island of competence for learning language" coincides with a period during which some special language faculty (which gets shut off in later life) is active, somewhat akin to critical periods for learning birdsong exhibited in some species. This in turn is taken as evidence in support of a Chomskyan-style language acquisition device.

Deacon feels that we need not appeal to such an explanation however, and that this phenomenon too can be used to support his theory of the emergence of language. He believes that the critical period for language acquisition can be explained in terms of features of infancy in general (noting also that in chimp "language" studies, it has been inadvertently discovered that immature chips acquire "language-like" capabilities such as symbolic representation [66], and perhaps some form of primitive syntax [39], more easily than mature ones).

He tries to tie this in with Elman's "Starting Small" hypothesis [31]. In essence, Elman was wanting to investigate Chomsky's thesis that languages were unlearnable because the data available to the learner may not be sufficient to uniquely determine the underlying grammar. This he did using

recurrent neural networks, and training them on a semi-realistic English-like artificial language. This language possessed some of the key properties suggested to make natural languages unlearnable, particularly multiple embeddings in the form of relative clauses. The task of the networks was to uncover underlying regularities in the structure of the training language and to use this to predict the next word in a given sequence. This, he suggests, is akin to the inferences a child must make when trying to elucidate the grammar of the language prevalent in their linguistic environment.

Initially, Elman trained his neural nets on corpora containing a whole range of sentences from the target language, of varying complexity, but found the performance to be very poor. So, in an attempt to understand where the networks were failing, he tried using corpora of increasing complexity. He started with a corpus made up entirely of simple, non-embedded, non-recursive sentences, and progressed through five different phases to a corpus containing the full range of complex constructions allowed by the grammar. This time he found that the networks were very successful in learning the training data, and were also able to generalise to novel sentences. However, this does not seem to be a good model for the circumstances under which children learn language: although adults do modify their speech when talking to young children, it does not seem that it can be guaranteed that children will hear no complex sentence forms until they have successfully acquired the grammar for more simple structures.

Elman notes that during the period in which children acquire their first language, they are also undergoing significant developmental changes, i.e. "if it is not true that the child's environment changes rapidly, what is true is that the *child* changes during the period he or she is learning language". In particular, he emphasises a gradual increase in memory and attention span, and attempts to model this within his networks by gradually increasing the access the network has to its own prior internal states. Starting with an initial memory window of 3-4 words, and increasing it in phases over the

course of the experiment, until in the fifth phase, memory was not limited at all, he trained the networks on the complete "adult" grammar, containing the full range of complex constructions from the outset. He found that, although the first phase needed to be much longer than in the previous experiment, similar degrees of accuracy were achieved, including the ability to generalise to novel sentences. Thus he concludes that in children, "limited capacity acts as a protective veil, shielding the infant from stimuli which may either be irrelevant or require prior learning to be interpreted". From the results of Elman's studies, Deacon suggests that immaturity itself may provide part of the answer to the observed "critical period" for language learning in human children.

Thus whilst Deacon is not dismissing the idea that human beings do not exhibit specific adaptations to enable them to learn language, nor that they may have evolved in some way as to make this task easier, he also introduces the idea that perhaps it is *languages* that have adapted to be more easily learnt by *us*. This is a very appealing notion, and has captured the imagination of various researchers.

2.6 The evolution of case markings

Finally, it is appropriate here to say a little about the evolution of case markings, as this is central to the topic of this thesis. Unfortunately, much as the literature on case systems is very extensive (and very contradictory), little has been written about the mechanisms by which case may have arisen. Generative linguists in the Chomskyan tradition believe it to be part of the grammatical endowment of the Universal Grammar, and thus to have evolved by whatever mechanism by which UG came into being. Conversely, Jackendoff believes it to be one of several complementary systems for the disambiguation of theta relationships that have been added to the grammar toolkit at

different stages in our evolutionary history. As touched on on page 35, he believes the eldest of these to be the use of word order to signal basic semantic relationships such who the participants are in an event being described and what their role in that event is. This facility, he suggests, have developed during the proto-langauge stage of langauge development. Case markings, and other methods for the dismbiguation of theta roles such as noun-verb agreement, he believes to be a later add on, which developed during the incremental transition from proto-language to fully syntactic modern language. One thing that seems to be significant about case and case-markers is that the literature on both seems to show very little consensus about a general set of cases or semantic roles and the precise function of each. This seems more in keeping with the idea that they are a nascent feature of the language itself, perhaps created by a consensus of the speakers themselves (see later a discussion of Luc Steels' work in this context [79]) rather than a feature of a genetically determined universal grammar.

Having reviewed the literature on theories of language evolution, I shall now go on in the following chapter to look at various attempts to model these theories using computational simulations, before describing Kirby's Iterated Learning Model [47, 49], on which the work in this thesis is based.

Chapter 3

Computational Models of the Evolution of Language

In Chapter 2, we discussed the issue of whether language learning is an ability innate to humans. That it is to some degree appears to be beyond doubt, but the question still remains exactly *how much* innate knowledge about the structure of language we are endowed with, and exactly what form it takes. Chomsky argues for the existence of a highly specified Language Acquisition Device, whilst simultaneously appearing to argue that this means that the language faculty could not have *evolved* by traditional selective processes due to the inherent complexity of such an organ. Other writers have suggested that the LAD could indeed have arisen by natural selection, or that such a highly determined structure is not necessary at all for the evolution of language.

Until recently it was possible to do little other than theorise, and to make sure that these theories were in-keeping with the existing data on linguistic structure and language acquisition. However, language is clearly a complex dynamical system, and such systems very rarely behave in the manner that

one might intuitively expect them to. With the advent of artificial life and the possibility of simulating complex behaviours on a computer, a new dimension to the field has arisen, resulting in a wide range of attempts to examine the cognitive pre-requisites for the evolution of language with the use of modelling techniques. In this chapter, I shall briefly review some of these approaches, which largely focus on whether a) language could emerge in the absence of an LAD or b) whether it is possible for an LAD to evolve by natural selection. I shall then go on to discuss the class of model that is central to this thesis, otherwise known as the "Iterated Learning Model" [47, 49], and to describe its relevance to the work presented here.

3.1 Language as an Emergent Phenomenon

One of the earliest models of the emergence of language was that devised by Luc Steels in the mid-nineties [72, 73, 75]. Steels believes that language is an emergent phenomenon which "spontaneously forms itself once appropriate physiological, psychological and social conditions have been satisfied" [76]. He has demonstrated that it is possible for populations of robotic agents to converge on a common vocabulary for objects in their environment simply by interaction in a series of negotiation rounds. Agents engage in "discrimination games" (whereby they must create distinctions between different objects in their environment) and "language games" (involving communicating those distinctions to other agents present). In essence, they pick out an object, assess how it is different from others in the environment, invent a "name" for it (assuming they do not have one already), and then use that name to communicate to others about the object they are referring to.

Agents are equipped with a number of sensory channels. Each channel yields a value in the continuous range 0.0 to 1.0, and it is the agent's task to build a "discrimination tree" for each, that will allow different objects in the

environment to be successfully distinguished. This is done by subdividing the sensory channel into a set of discrete categories. For example, a channel measuring the intensity of visible light might initially be subdivided into categories "bright" ($0.5 \leq x < 1.0$) and "dim" ($0.0 \leq x < 0.5$). The agent will distinguish objects in its environment by assigning to each a unique featureset. Thus it must ensure that there are sufficient perceptual categories to enable this. If a new object is encountered for the which a unique featureset cannot be created, the agent must further refine the distinctions it is able to make by adding new branches to the discrimination tree for a particular feature, for example, subdividing the range for "bright light" into "quite bright" and "extremely bright". Initially agents have no discrimination trees at all and all channels are completely unsegmented.

A successfully distinguished object is assigned a name based on its unique featureset. An entry is added to the agent's internal lexicon recording the appropriate name-featureset pairing for future use, and then agents attempt to communicate with each other about the objects in the environment. The "speaker" identifies an object as the topic of the interaction and makes it known to the "hearer". Both agents identify the object by retrieving a featureset for it which cannot be applied to any other object in the environment – the featuresets identified by each agent need not be identical. The "speaker" uses the featureset is has retrieved to look up a "name" for the object, and then communicates this to the "hearer". The "hearer" will look up this name in its own lexicon to see if it corresponds to the object in question: if it does, then communicative success has been achieved. If the featureset associated with that word in the hearer's lexicon does not match the object being identified, then this is considered a communicative failure. Alternatively, the word may be one with which the hearer is unfamiliar, in which case it will add it to its lexicon, paired with the unique featureset that it has previously identified for the object (which may be a different featureset to that associated with this particular word in the speaker's lexicon, and which may or may

not lead to future communicative failures, dependent on the introduction of other objects with similar characteristics).

In these experiments Steels has demonstrated that agents can successfully discriminate and communicate about objects in their environment, performance in both these tasks increasing with the number of "games" played. The discrimination trees created by each of the agents in the simulation become increasingly similar as time goes on: requiring agents to be able to communicate with each other causes the commonality of the environment in which they exists to become reflected in the representations of that environment that each holds.

Steels has also done a variety of similar studies in which agents engage in dialogue in order to identify each other by their relative positions [74], or attempt to communicate about structured cognitive memories [77], once again demonstrating how agents which share a common environment often develop similar internal representations of that environment, the degree of similarity being increased significantly when agents are required to engage in communicative exchanges about those representations. He refers to this process as "structural coupling". This occurs when an agent conceptualises reality in terms of its internal representation, and then verbalises the conceptualisation; to understand the verbalisation, another agent is forced to adopt/hypothesise similar conceptualisations leading to similarities in its internal representation of the environment. Throughout all his simulations, Steels has managed to demonstrate the spontaneous emergence of a coherent lexicon amongst a population of agents, as well as combinatorial structure in the utterances of agents which mirrors the structure of the environment in which they find themselves (for example a lexical entry for the property *red* may emerge and be used in conjunction with entries for quite different objects, both of which happen to be red). He claims that this is significant in the development of syntax, but as yet has not succeeded in getting agents to express predicate argument relationships.

To a certain extent, the work of John Batali follows on from where Steels leaves off. Batali's work investigates whether agents using a similar "negotiation" type paradigm can develop co-ordinated systems for conveying structured meanings [3]. His agents are recurrent neural networks, and they exchange sequences of tokens intended to represent meanings within a particular semantic space. This space is made of 10 "predicates" such as *happy, sad, excited* (which are all single-place predicates) and 10 "referents" which can best be described as a complex pronoun system. This is derived from an English-based creole and includes the standard English pronouns *me, you* and *they, one* in the sense of a generic third person singular, *we* as a group of speakers, *yall* as a group of hearers and *all* referring to a group that includes both speaker, hearer and others, plus additional pronouns *mip, yup* and *yumi*, which refer to a group which includes the speaker, a group which includes the hearer, and a group including both speaker and hearer but no others, respectively.

The combination of 10 predicates with 10 referents in this way gives a space of 100 possible meanings. Agents represent these meanings by means of a vector which can hold ten real-valued numbers between 0 and 10. Six of the values in the vector determine the predicate and four designate the referent. Each of the 100 meanings in the meaning space is represented by a predefined sequence of 0s and 1s.

Agents engage in rounds of "negotiation" where they attempt to communicate with each other and learn from the utterances made. An agent is chosen at random from the population to act as learner. Ten further agents are chosen in succession to "teach" the learner. Each teacher sends a set of sequences, each presented once, in random order. These are represented by setting each value in the meaning vector to either 0 or 1, according to the meaning to be conveyed. The speaker then runs through each of the tokens in its repertoire and inputs them to its network, selecting the one which gives the output closest to the value in the meaning vector. If necessary, further

tokens are selected to modulate the first until an output close enough to the correct meaning is achieved. The hearer processes each of the tokens in the sequence in the order in which they are presented, and compares its own output with the meaning intended by the speaker (any value within 0.5 of the value in the speaker's meaning vector is considered to be correct). Adjustments are then made to the network by back-propagation with the aim of learning the correct pairing between sequence and meaning.

Initially, the utterances produced by the population are completely uncoordinated. Agents send very long strings of seemingly random tokens which suggests that their attempts to find a sequence which has the correct meaning are not very successful. However, after very many negotiation rounds, Batali claims that "agents develop highly co-ordinated communication systems that incorporate structural regularities reminiscent of those in human languages", with a high level of communicative accuracy and short, distinct utterances for each meaning. Significantly, he claims that systematic regularities exist between the meaning patterns and the sequences that convey them, although they are not *completely* regular. He proposes a quasi-linguistic analysis of the language(s) developed by his system as a root which expresses the predicate plus some modification to the root that expresses the referent. The referent does not necessarily take the form of a separate "word", and may appear as a modifier element, interspersed with the characters that make up the root.

Thus, both Steels and Batali have succeeded in demonstrating the spontaneous emergence of language-like behaviour in populations of agents that are required to communicate with each other about aspects of their environment. In neither case do agents possess any kind of "Language Acquisition Device" or indeed any language specific knowledge. However, the behaviour exhibited is very simple and there is little to suggest how language might have progressed from these simple stages to the complex phenomenon that it is today.

3.2 Modelling the Language Acquisition Device

The following models take almost the opposite stance to those discussed above: rather than trying to demonstrate what aspects of language can emerge in the absence of a Language Acquisition Device, they focus upon modelling the evolution of such a language faculty, in order to discover whether such a thing *could* indeed arise by selective processes, in particular whether it is possible to demonstrate it evolving to increase the ease of acquistion of languages in the an agent's environment.

Simon Kirby and Jim Hurford [48] have investigated the relationship between linguistic selection for languages that are more easily parsed and natural selection for agents which are better able to acquire their target language, in order to discover whether either or both of these forms of selection are sufficient to cause the evolution of a Language Acquisition Device. Using a principles and parameters type framework, they demonstrate that selection for an LAD seems only to occur in tandem with linguistic selection.

Their simulation tracked the emergence of an LAD amongst a population of interacting, reproducing software agents. Each agent is made up of a grammar and a genome. The grammar is represented as an 8-bit string of 0s and 1s. This gives a total of 256 logically possible languages, although Kirby and Hurford aim to show that due to the actions of natural selection and linguistic selection, the actually occurring languages will not be evenly distributed within this space, just as natural languages are not evenly distributed within the space of logically possible languages. The genome of each agent is also an 8-character string, and each position on the string can be one of three possible alleles – either 0, 1 or ?. This, the genetic makeup of an agent influences the possible range of languages it can learn: if the value of any given bit on the genome is set to 0 or 1, then the value of the corresponding position on the agent's grammar is predetermined to be a 0 or 1 respectively, and the

agent will only ever be able to acquire grammars with the appropriate value in this position. Thus "genes" with a value of 0 or 1 can be thought of as representing grammatical principles.

A value of ? in the genome represents a grammatical parameter whose value is set when a language is acquired. Thus an agent with all the positions in its genome set to 0 or 1 can be thought of as having no parameters, only principles: a fully nativised grammar. No learning is required in order for it to acquire a language. Conversely, an agent with only ?s in its genome is fully plastic and unconstrained with regard to the range of languages it can learn. It has no grammatical principles.

The population of agents is made up of both adults and learners. The adults produce utterances from which the learners attempt to acquire the language of the current population. Utterances are encoded as an 8-bit string of *s with either a 0 or a 1 at one position. Using a slightly modified version of Gibson and Wexler's Trigger Learning Algorithm (or TLA) [34], the learner attempts to analyse the sentence by comparing it with the current value of its internal grammar. If the sentence "fits" the grammar, i.e. the 0 or 1 value in the incoming string corresponds to the same value in the appropriate position of the learner's grammar, then the string is considered to have been successfully parsed, and the learner's grammar remains unchanged. If however, there is no fit, then the learner chooses one parameter, at random, from the those which are not predefined by its genome, and resets it. It then attempts to re-process the failed sentence. If this time it is successful, the change to the grammar is preserved. If it fails, then the original parameter value is retained, and the process is repeated.

A critical period is employed, during which learning occurs. After this, agents become adults and cease to be able to modify their internal grammars. At this stage, the success of language learning is measured: the newly mature adult agents interact with each other, and are scored on their ability to

successfully process incoming utterances, as well as their ability to produce utterances that are successfully processed by other agents. This score is used to assign a fitness to each agent based on its communicative ability. In addition to this, 10% of an agent's utterances as an adult are rated on how easily parsed they are. In order to reflect the idea that some parameter settings would result in grammars that are more easily parsed than others, parsability was measured by an arbitrary function that prefers 1s in the first four bits of the vector. Once the fitness of the individual agents has been ascertained, those in the top 90% of the population are selected to breed, thus passing on the principles encoded within their genomes (rather than the final settings of their grammars). Reproduction involves 1 point crossover of the genomes of two individuals, plus mutation at a probability of 0.001 per allele.

The simulation was carried out in two phases: in the first, the role of natural selection alone was examined. The original population used was of 100 agents with fully plastic LADs, that is, whose genomes contain eight ?s. Under these conditions, the fitness of the population never reaches its maximum level, and although agents receive a higher fitness score if the language they acquire is more easily parsed, (i.e. if it has a greater number of 1s in the first four bits of its grammar), this does not seem to result in the nativisation of 1s in these positions. Some parameter settings were nativised, but these appeared to be entirely random and not necessarily those that would constrain the learner to be able only to acquire a more easily parsed grammar.

In the second part of the experiment, an element of *linguistic* selection was introduced, by modifying the parameter setting algorithm to favour "trigger sentences" from more easily parsed grammars. In 10% of cases, the learning agent will only keep the new parameter setting if the new grammar has a higher parsability than the old one. The results of this phase are quite different to the first: the population converges very quickly, and the languages converged upon rapidly evolve towards those which are more easily parsed. Natural selection of agents themselves is also observed, although at a much

slower rate, resulting in ? alleles in the first four positions of the genome vector gradually becoming 1s. Thus, the LAD does eventually evolve to at least partially constrain learners to languages that are more functional.

Kirby and Hurford conclude that selection pressure on the individual to be able to interpret and speak more easily parsed languages is not in itself sufficient for the learner to become constrained by natural selection to *only* be able to acquire such languages. This is because the most important thing to the learner is to achieve sufficient fitness to be able to reproduce by being able to correctly learn the language of its speech community. Thus, if the community speaks an optimally parsable language, then a mutation that constrains the learner to only learning parsable languages will not really give much of a reproductive advantage – this learner may acquire the target language more easily than those around it, but they too are obviously able to acquire it perfectly successfully, or it would not be the prevalent language. On the other hand, if the language of the speech community is less than optimally parsable, then any mutation which constrains the learner to be able to learn only more parsable languages would result in that learner being unable to acquire its target language, and thus a paradoxical reduction in fitness.

Thus, they argue, "there is no way in which a mutation that increases the functionality of the LAD (in the sense that it constrains languages to be parsable) can give a direct fitness advantage to an individual ... even though the fittest population would be one that possessed just such an LAD."

However, in the situation where agents favour languages which are more easily parsed, even without natural selection of agents, speakers converge on those languages which are optimal. Thus, claim Kirby and Hurford, it is simple for an LAD to gradually evolve which mirrors the existing constraints on variation. A mutation which prevented an agent from acquiring a less easily parsed language would no longer be likely to carry a penalty for the

individual, as the language of its community is unlikely to be such a sub-optimal language. This result appears to add weight to Deacon's suggestion [27] that languages may evolve to fit the biases imposed by the learner.

Ted Briscoe has also done a series of similar experiments, attempting to demonstrate the co-evolution of language and the language acquisition device [13, 14, 15, 16] using "grammars" with a more explicity linguistic basis than the rather abstract model described in Kirby and Hurford's experiments. He is also able to demonstrate the evolution of languages themselves towards greater functionality (which he measures in terms of learnability, interpretability and/or expressivity rather than parsability), with simultaneous evolution of the Language Acquisition Device to aid the acquisition of these more functional languages.

The agents in Briscoe's experiments have a grammatical framework based on a Generalised Categorial Grammar (which is intended to represent Universal Grammar). This is augmented with a series of 20 parameters specifying variable elements within the framework, such as the positions of arguments relative to their functors (which he calls *gendir*) or the relative position of subject arguments (*subjdir*), or the requirement for a verb in the second position of the sentence (*V2*) as is observed in languages such as German. These parameters have a partial ordering: for example, *subjdir* is more specific than *gendir*. Each parameter has two possible values (in the case of the *gendir* and *subjdir*, left or right, and in the case of *V2*, true or false), and the value of the parameter can be specified as an Absolute (i.e. cannot be changed by learning, but can be regarded as a principle of the underlying UG), Default (i.e. can be reset as part of the learning process) or Unset (i.e. has no initial value). This framework defines about 300 different grammars. However, not all of these are stringset distinct – some are subsets of others; in total there are about 70 distinct full languages, many of them resembling the syntax of clearly attested human languages.

Agents in the simulation parse and generate sentences which are compatible with their current parameter settings, as in Kirby and Hurford's simulations. If, during the learning phase of development, they are unable to parse a particular sentence, then they attempt to update their parameter settings to cover it. Briscoe's original learning algorithm [13] is also based on Gibson and Wexler's Trigger Learning Algorithm [34] although with a few important differences:

- An element of memory has been incorporated, as Briscoe argues that the memoryless nature of Gibson and Wexler's algorithm, in which a learner may continually set and re-set the same parameter, is psychologically implausible – there is no evidence that children blindly revisit previous hypotheses before eventually converging on a target. Instead, the algorithm is changed so that each parameter can be reset only once during the acquisition process.[1]

- The standard TLA allows only one parameter to be reset per trigger. Briscoe's algorithm allows n resettings, where $n \leq 5$; the exact value of n is subject to alteration by the pressures of natural selection.

- The standard TLA chooses parameters at random for resetting in response to an unparsable trigger. Briscoe's algorithm starts with the most general in terms of the hierarchy defined by the partial ordering of parameters. Thus the gendir parameter would be set before the subjdir one.

[1]One might argue that this is equally psychologically implausible: whilst not "blindly revisiting" previous hypothesis, children do sometimes appear to "backtrack" during the process of acquisition. For example, it is not unusual for a child to be able to productively use irregular verb forms such as *was* and *went* before past tenses of regular verbs have been learnt. Once the regular forms have been acquired, they may go through a period of "regularising" irregular verbs before eventually getting the right mix of regular and irregular [32].

However, in his later experiments [14, 15, 16], Briscoe has changed his learning algorithm to give it a Bayesian basis. Whereas previously, a single occurrence of a particular trigger would cause parameter resetting, according to the revised algorithm, it simply increases the confidence with which the agent believes a particular parameter setting to be correct.

Once learning is complete, adult agents continue to interact and are assessed by means of a fitness function, whereby the ability to communicate via language confers a selective advantage, and the ability to communicate by a more learnable, more expressive and/or more interpretable variant language confers a further advantage still. Those agents with a higher than average fitness are selected to reproduce, whilst those with lower fitness may die prematurely. Reproduction proceeds by passing on of the *initial* parameter settings of the agent, not the learned ones, and involves one-point crossover of agent genomes with a probability of 0.9, plus single point mutation with a probability of 0.05.

In the first set of simulations, agents were not selected for on the basis of communicative ability, but were allowed to reproduce regardless of whether they had successfully acquired a functional grammar. The initial population of agents was a genetically invariant one, but the linguistic environment was kept continuously heterogenous, thus providing variation which might act as the substrate for linguistic selection. This was achieved by two means:

The first is by migration, in the form of additional agents (about one third of the population) speaking a second full language introduced at periodic intervals. This was done at such a rate that there were always two languages present in the population, yet communicative performance was kept at a level of at least 90%. The second language was never more than three parameter-settings away from the original.

With this method of introducing linguistic diversity, when there is no natu-

ral selection for agents, linguistic selection still occurs – populations tend to converge on those languages which are more easily learnable. Without pressure for expressivity, the population shows a strong tendency to converge on subset languages, i.e. those with sub-optimal expressivity.

A further element of linguistic selection was then added. The fitness of agents was measured by assessing the interpretability and expressivity of their language, and this was used to determine which speakers will provide the trigger stimuli for the next generation (although all agents still reproduce and pass on their initial parameter setting to the new learners). This causes agents to stop converging on the simpler subset grammars, but instead to favour easily learnable and interpretable *full* languages.

In further simulations, rather than using migration to introduce linguistic diversity, Briscoe used a bilingual initial population made up of two genetically identical adult groups. These groups speak different full languages which contrast in terms of their learnability and/or their interpretability, for example a language whose underlying word order is Subject-Object-Verb (SOV), and an identical Subject-Object-Verb language with a V2 requirement (SOVv2). The SOVv2 language is slightly easier to interpret because of the superficial Subject-Verb-Object ordering found in unembedded sentences, which requires a slightly lower working memory load.

Without natural selection of agents, linguistic selection tends to favour the SOV language, because it requires one less parameter to be reset, making it easier to learn. Thus this language quickly dominates. However, adding a fitness function as described above to include pressures for interpretability and expressivity results SOVv2 to becoming the dominant language. In both cases, there was no significant change in average agent fitness.

Finally, simulations were run in which the full fitness function is used to determine which agents would be able to *reproduce* and pass on their orig-

inal parameter settings. The effects of this were investigated in a relatively constant linguistic environment, i.e. one where there were no population migrations, and also in a continuously changing linguistic environment with regular migrations as described previously. In both cases, the original population was of genetically identical agents speaking one full language.

When there are no migrations, convergence on parameter settings which enhanced the learnability of the dominant language was seen. (It is worth noting that even in this relatively constant linguistic environment, there was still some heterogeneity, due to the occasional misconvergence of a learner, for example). This usually takes the form of an increased number of default parameter settings and a reduced number of unset parameters. There is also a tendency for the value of n (i.e. the number of of parameters that are reset when a trigger is found to be unparsable) to be reduced to 2 or 3. The reduced learning costs associated with the evolution of the population resulted in an increase in agent fitness, although interpretability, expressivity, and communicative performance tended to stay the same. This is a clear example of genetic assimilation – agents evolving to be able to acquire the dominant language more effectively. The low level of linguistic variation creates very stable selection pressures for genetic change to be based on.

Adding migrations, by periodically replacing approximately 30% of the population with agents speaking a second full language, there is a lot more linguistic variation, and the dominant language changes rapidly. It does not confine itself to the two languages originally present in the population, either, but many different languages are sampled. However, as before, the LAD still evolves to improve learnability, although instead of replacing unset parameters with default values, as before, the tendency is to create grammatical principles as well as defaults. Once again, an increase in fitness was observed due to the decreased learning costs brought about by the evolution of the LAD, but also this time, increased parsability was observed.

When discussing these results, Briscoe suggests that in a rapidly changing linguistic environment, one might expect that nativisation of default parameters would be more common than nativisation of principles because a correct principle which subsequently became incorrect due to change in the dominant language would incur a very high cost, whereas a default parameter setting which became incorrect would only incur the same cost as a parameter that had been left unset. Whereas conversely, in a relatively constant linguistic environment, the nativisation of principles incurs no cost as they are unlikely to become incorrect, although there is relatively little pressure for principles rather than correctly set defaults. However, the observed results seem to show the opposite trend. Briscoe suggests that this is due to the linguistic selection imposed by the nativisation of a grammatical principle: any languages that do not obey this principle effectively become unlearnable. Although in a relatively stable linguistic environment, a grammatical principle confers very little advantage over a correctly set default, when the environment is changing rapidly perhaps the nativisation of principles makes it easier for a population to converge on one single language by constraining the space of possible languages that can be learnt.

So, in summary, while Kirby and Hurford have demonstrated that in the absence of linguistic selection, natural selection seems powerless to steer the Language Acquisition Device towards that acquisition of "superior" languages, both they and Briscoe appear to have found that when linguistic selection and natural selection act in tandem there is significant convergence on parameter settings that aids the acquisition of more functional languages. Genetic assimilation does appear to be occurring, despite Deacon's assertion that the rate of linguistic change is far too fast to provide a stable base for this to happen. In fact, if anything, more drastic change seems to occur when there is *more* linguistic change, as seen in the nativisation of principles in populations undergoing frequent migrations. Thus, although Briscoe recognises that the time taken for grammatical change and for biological evolution

depends on many factors such as the population size, geographical dispersal, the diffusion rates of genes and of variant grammatical forms etc., he believes that his results demonstrate that genetic change can occur even in the face of very rapid linguistic change. This, he says, is because the space of possible grammars is vastly larger than the number of possible grammars which can be sampled by the population in the time it takes for one default parameter or principle to become prevalent within the population. Typically, he claims, 5% of the grammatical space may be sampled in such a period, which leaves 95% of the selection pressure for genetic assimilation constant at any one time.

However, there is one question that might be raised by this. If genetic assimilation can occur so rapidly and with such ease, what stops it from happening *within* linguistic communities? It is not generally suggested that different human populations with different predominant languages have genetic differences that enable them to learn their own languages much more easily than that of another speech community. An infant brought up in a different speech community from that of its parents is not generally impaired in its language acquisition. Yet, if genetic assimilation occurs as readily as demonstrated by Briscoe's results, would this not be the case?

It is perhaps necessary to interpret Briscoe's results with caution. In his simulations, mutation involves a simple change to a binary valued parameter, occurring with a probability of 0.05. Every mutation brings about a change in the functionality of the LAD it specifies and mutations never result in an LAD that is non-functional, or unable to acquire a language of any sort. By contrast, the probability of any given human gene mutating during the lifetime of an individual is 1 in 2500. However, the human genome is so large that one would expect about 10 mutations in the passage of genome from parent to child [40]. Without knowing exactly what proportion of the genome is devoted to language-specific information, it is difficult to judge whether Briscoe's mutation rate of 0.05 is realistic or not. Regardless, at whatever

rate mutations do occur within the part of the human genome specifying language capabilities, this does not equate to an equivalent rate of principle nativisation or parameter flipping. A single mutation to a single gene in the language faculty would almost certainly not result in the binary change in behaviour that Briscoe's model encapsulates. Firstly the genetic code is not binary, but quarternary. Thus a change to one base in a DNA sequence yields not one but three possible outcomes. Secondly, such a mutation may well have no structural effect whatsoever on the protein being encoded by the gene in which it has occurred, and no structural effect on the protein means no behavioural effects in the individual. Thirdly, if any changes in protein structure do occur, it is very unlikely to result in parameter flipping or principle fixation, but simply to render that protein dysfunctional. Finally, many of the complex characteristics of human biology are controlled not by one single gene, but by a number of genes. Thus it seems likely that for a parameter's default value to be flipped, or for a principle to become fixated within the LAD, it would probably require a number of specific mutations to a number of different genes. Thus it seems that the mutation rate in Briscoe's simulation is actually very high. By contrast, the mutation rate in Kirby and Hurford's study described above [48] is set to the much more conservative rate of 0.001.

Of course, the whole simulation is a simplification of all the factors involved, including of the language being spoken, the process of its acquisition, as well as the way in which the language faculty is represented genetically. Such simplifications are necessary in order to be able to make such a model viable, and as such, they do not necessarily undermine his results in any way. However, it does serve to highlight that we should be careful in drawing any *firm* conclusions from them: if the dynamics of genetic change have been wrongly estimated *relative to* the dynamics of language change then it is not necessarily possible to state that because genetic assimilation *does* occur in the model, that it could *also* have occurred in evolving populations

of pre-linguistic hominids. It is interesting to note that although Briscoe's model of language is far more complex than that used in Kirby and Hurford's simulations, he allows genetic mutation to occur with twenty times higher probability.

However, with this caveat in mind, both Briscoe's and Kirby and Hurford's models appear to show that, under the right circumstances, the Language Acquisition Device can indeed evolve to aid the acquisition of language and to constrain the range of possibly languages that can be acquired. However, the models used in both studies incorporate a very sophisticated framework for language acquisition and on which selective processes will act, without explanation for its existence. Thus it has been shown that with some form of genetically determined language specification in place, selective pressures do exist to further constrain the range of learnable languages, but what has not been demonstrated is *how* and indeed *whether* this complicated blue-print might have evolved in the first place.

3.3 Iterated Learning Models

The final class of model I will discuss here, known as the "Iterated Learning Model" (a term coined by Henry Brighton [12]) was developed by Simon Kirby, [47, 49] and is intended to demonstrate that a Language Acquisition Device is *not necessary* for the emergence of some of the crucial features of human language (namely compositionality and recursion). Instead Kirby proposes that these features could in fact be an inevitable outcome of the dynamics of language transmission. Iterated Learning Models are so called because they simulate an iterative learning process over a series of generations of agents. Agents learn observationally from the behaviour of others in their environment. Through this process of cultural transmission, structure emerges spontaneously.

An overview of the model is as follows (although I will gloss over the details here, as I will be describing my own implementation of a similar model in Chapter 4): the simulation consists of a population of identical agents, initially with no language knowledge. In the experiments Kirby describes in his early studies, the population size is simply 2: one learner and one speaker. Agents are equipped with a simple learning algorithm with which they can make generalisations about the linguistic behaviour they observe. The speaker produces utterances intended to represent items from a predefined meaning space, which are conveyed to the learner as string-meaning pairs. It does this by consulting the grammar it has induced from the utterances to which it was exposed when it was a learner. If it is unable produce an utterance for a given meaning, it will invent one. (The very first speaker in the simulation, having no knowledge of the language at all, will have to resort to invention for every meaning that it wishes to convey). The learner receives the speaker's utterances, and uses them to build its own grammar for the language being spoken. After a fixed number of interactions of this type (in Kirby's simulations, 100),the speaker is removed from the population, the learner becomes the new speaker, and a new learner is introduced.

Grammars are built by extracting regularities from the utterances to which agents are exposed. The first stage in this process is to build a single grammar rule for the incoming utterance to relate the string heard to the meaning intended. The second part of the grammar induction process is to try and make generalisations between the newly incorporated rule and other rules in the grammar. To do this, Kirby uses two basic operators, *chunk* and *merge*, plus a set of heuristics to determine when each of them should be applied. The operators are based directly on a grammar inducing algorithm by Stolcke [80], although Stolcke himself uses a Bayesian technique to determine when they should be used.

The main function of the *chunk* operator is to attribute parts of the string to elements of the meanings. Thus rules are compared on a pairwise basis, and

if two rules are found to differ by only one part of their meaning, and the strings associatiated with those rules also differ by a single substring, then the difference in the meaning is attributed to the differences in the strings. For example, if rules exist that state that the string *j,o,h,n,l,o,v,e,s,m,a,r,y* has the meaning *loves(john, mary)* and that the string *p,e,t,e,l,o,v,e,s,m,a,r,y* has the meaning *loves(pete, mary)* then the agent will conclude that *john* is attributable to the substring *j,o,h,n* and that *pete* is attributable to the substring *p,e,t,e*. The two original rules are removed from the grammar, and new rules are created to reflect this.

The *merge* operator is intended to remove redundancy from the grammar. Thus when pairs of rules are compared, if two are found to be equivalent, the second will be removed from the grammar. The reader is referred to page 95 for full details of the implementation of these operators.

The crucial part of the model is the transmission of information *across generations*: after a pre-specified number of utterances, the speaker is removed from the simulation, the learner becomes a new speaker, and a new agent with no linguistic knowledge at all is added as the new learner. Thus the previous learner will pass on the language that it has learnt to this new agent before the process is repeated yet again.

In Kirby's original studies [43, 46], agents are required to produce utterances for a range of meanings made up of five "actions" (all of which require two participants) such as *loves*, *hates*, etc. and five "individuals". These are combined to give a total of 100 possible meanings to be expressed (as no "individual" can be both subject and object of an event being described). As a result, it is very unlikely that any given learner will get to observe utterances for *every* item in the meaning space, as meanings are selected at random, *with replacement*. This means that in all probability there will be at least some meanings that occur more than once, and some that do not occur at all. Under these circumstances, agents early in the simulation tend to

have entirely holistic grammars in which the whole of a complex meaning is expressed by an unanalysed arbitrary sequence of characters. At this stage, only a very small proportion of the meaning space can be expressed, and grammars are very large – there is generally one rule per utterance that has been observed. As the simulation progresses, however, generalisations start to emerge, resulting in a transitional phase of semi-compositional behaviour, during which the proportion of the meaning space that can be expressed rapidly increases, although the size of the grammar remains large. After this, agents go on to converge on very tidy, minimal, fully compositional grammars which are able to express the full meaning space. Two separate non-terminal categories are generally seen in these grammars, one used to express the *individuals* from the meaning space, and the other the *actions*, thus encoding the noun/verb distinction. A single top level rule for these categories specifies the order in which they are to be combined, thus determining which of the "words" of the "noun" category are the subject and object of the sentence.

In further work, Kirby added a set of five "actions" which he calls embedding predicates – notions such as *believes* and *knows* [44, 45] – broadening the meaning space to include ideas such as "John knows Pete loves Mary". He employs a paradigm akin to that used in Elman's work on the "Starting Small Hypothesis" [31], gradually increasing the degree of embedding in the meanings that each agent is required to express during its time as a speaker. In this way he has succeeded in demonstrating the emergence of a fully recursive grammar. This includes a second verbal category used to denote the embedding predicates described above, and two rules for the construction of sentences: the first specifies that a sentence may be made up of one word of the verb category and two of the noun category (in the appropriate order) and the second says that it can also be made of a noun and a verb, *plus another sentence*.

In yet another study, a non-uniform meaning space was employed [47]. In this case, agents were biased to favour shorter strings, and were requested to

express certain events from the total space of meanings much more frequently than others. A pattern of stable irregularity was the result: agents tended to favour short holistic strings for the commonly occuring meanings, but still used longer compositional utterances for those that were less frequent. This behaviour clearly mirrors the tendency in natural languages for the most commonly used verbs to be irregular, and to have morphology that is not strictly governed by the rules of that language.

Kirby claims these results are a consequence of the "dynamics of language transmission". In short, it is due to what he refers to as the language bottleneck: because agents cannot hope to sample utterances covering the entire meaning space during their lifetime, any language in which the meaning of a string can be easily predicted from the its structure will stand a much better chance of being successfully propagated from one generation to the next. Initially, in these simulations, such languages clearly do not exist: the grammars of the initial agnets are almost purely holistic and amount to little more than vocabulary lists relating the whole complex meaning to single, non-decomposable string. However, *if* similarities between strings with similar meanings should occur *by chance*, and if the appropriate generalisations are made, this may well result in selection for compositional structure. If the meaning space were smaller or agents were in some way guarenteed to hear utterances covering every meaning in it during their lifetimes, then a compositional language would have no selective advantage. Chance similarities between strings might occur, resulting in the kinds of generalisations seen here, but they would be no more learnable than holistic rules, thus they would not be selected for in any meaningful sense. This is mirrored in Kirby's later study on stable irregularity [47] in which rules to create utterances for meanings that occur most frequently tend to remain holistic.

One thing that is very important is the caveat mentioned above: *"if the appropriate generalisations are made"*. These generalisations depend on the learning algorithm employed, which as Kirby says, must be "able to exploit

pattern, or [be] biased towards generalisation" [45]. Naturally, this is a very important pre-requisite, for an agent that is incapable of making the correct generalisations will not be able to learn a compositional language, and if a compositional language cannot be learnt, then it certainly will not emerge spontaneously from a holistic one. The induction algorithm employed in these experiments was based on one designed for learning natural language [80], which essentially looks for the kind of similarities between strings that one might expect from a compositional language: in short, it is *looking* to associate parts of the string with parts of the meaning. Agents that sought to make other kinds of generalisations would probably not result in the emergence of the compositional behaviour seen here. Similarly, given the biases inherent in the learning algorithm employed here, it would not be possible for *any other* type of language to have arisen. Thus, whilst as Kirby rightly claims, agents are capable of learning holistic grammars just as well as compositional ones, they are strongly biased in favour of the latter. This has lead critics of his work to suggest that whilst his results are certainly striking, they are to some degree inevitable.

Nonetheless, what Kirby has been able to demonstrate is the spontaneous emergence of important features of natural language, namely compositionality and recursion, in the absence of a Universal Grammar, or any other "language blueprint". In some respects though, the biases of the learner are fulfilling the same role as a Universal Grammar, by constraining the types of languages an agent can acquire.

In order to address the criticism that the learning bias of the agents in his simulations is too strongly in favour of compositionality, Brighton and Kirby have done a series of studies described in [12, 11] whereby an alternative approach is employed – that of the Minimum Description Length (MDL) principle. MDL, they claim, rests on a solid mathematical justification for induction, whereby the best hypothesis for some observed data is considered to be the one that minimises the sum of a) the encoding length of the

hypothesis itself, and b) the encoding length of the observed data in terms of that hypothesis. Hypotheses regarding the grammar that produced the utterances that an agent has observed are expressed in terms of Finite State Unification Transducers. For each utterance drawn from the language to which the agent is exposed, a pathway through the transducer is created which associates the string with the its meaning. The result of this is known as the "prefix tree transducer" and acts as the starting hypothesis for the grammar. The agent then attempts to reduce the MDL of the grammar, by merging states and edges of the transducer based on commonalities between meaning and strings. It does this on a hill climbing basis, whereby all possible merges are evaluated in turn to see which results in the greatest reduction of the MDL, and that operation is chosen. This process is repeated until no further reductions can be made resulting in a "compressed transducer".

In general, it is the case that for a holistic language, a prefix tree transducer results in the lowest MDL, whilst for a compositional language, it will be a compressed transducer. In the case of a holistic language, structure is not related to meaning in any way, and thus there are no generalisations that can be made, whereas in the case of a compositional language, there is a typical structure to the transducers learnt, whereby each feature is dealt with by a separate fragment of the transducer. After the constituent part of the signal has been parsed, its meaning is logged by that fragment, and the transducer moves on the next constituent. After all the constituent parts of the signal have been parsed, the union of the logged meaning fragments is used to generate the whole meaning.

Brighton and Kirby then conducted experiments in which they assessed the stability of various language types, by artificially constructing both holistic and compositional languages, and presenting a subset of the strings from one such language to the agents described above. They then measured the expressivity of the languages that these agents had learnt, by calculating what proportion of the meaning space they would be able to express without re-

sorting to invention. Unsurprisingly, for holistic languages, the expressivity was directly proportional to the proportion of the meaning space observed during the learning phase. In order for 100% expressivity to be achieved, 100% of the meaning space must be sampled. For compositional languages though, 100% expressivity is achieved after a relatively small number of presentations, as an agent does not have to have observed all possible strings from a language to be able to express them all, once appropriate generalisations have been made.

Brighton and Kirby then went on to investigate how the structure of the meaning space effects the stability of these languages types, and in particular to assess the relative stability of compositional languages to non-compositional languages. In order to do this, they created a variety of different meaning spaces, varying either the number of dimensions in the meaning, or the number of values each dimension could take, and repeated the experiment described above. As before, they found that when only a small proportion of the meaning space was presented to the agent during language learning, the relative stability of compositional languages was far greater than that of holistic languages, for the reasons discussed above. This decreases as the proportion of the meaning space observed during learning is increased. However, what they also discovered, was this advantage increased with the complexity of the meaning space: this effect was noted both for experiments where number of dimensions from which meanings were composed was increased, and also those where the number of values these dimensions could take was increased, in both cases resulting in much larger meaning spaces. A further experiement was performed, in which the overall size of the meaning space was held constant, but the number of dimensions and the number of values of those dimensions were varied together. This resulted in, at one extreme, meaning spaces with very many dimnensions, each of which could take only a very few different values, and at the other, meaning spaces with very few dimensions with many possible values. The results of this

showed that once again, when only a small porportion of the total meaning space is presented during learning, compositional grammars show a much higher relative stability than holistic ones, and that this effect decreases as the proportion of the meaning space presented during learning is increased. However, what was also shown was that the relative stability of compositional grammars is greater still for meaning spaces in which there are many dimensions with only few possible values.

Thus Brighton and Kirby conclude that the stability of compositional languages when being transmitted from one generation to the next through a population, and thus ultimately the likelihood of compositionality occurring is greatest under conditions where there is poverty of stimulus, i.e. agents are only exposed to a small proportion of the total meaning space during language acquisition (referred to as the *learning bottleneck*) and also where the meaning space is highly structured, in particular where there are a great many different dimensions to the meaning. These features, they argue, correspond to conditions specific to hominids and could well have been instrumental in the emergence of compositional language.

Kenny Smith [70] has furthered this investigation into the influence of meaning space structure and the learning bottleneck on the emergence of compositional behaviour in a full iterated learning model (as opposed to Brighton's model which compares the stability of langugaes over a *single* generation of cultural transmission). Smith's agents are implemented as a basic associative network. Initially the weights on the networks are set to 0, and when learning occurs, they are updated according to some weight-update rule W, the details of which I shall not go into here. The population dynamic is such that there is only ever one agent at any given point in time. The current agent generates a set of meaning-signal pairs by applying the network production process to every meaning in the environment, and is then removed from the simulation. A new agent enters the simulation, again with initial weights set to 0, and using the update rule W. This agent receives e exposures to signal-

meaning pairs produced by the preceeding agent. After each signal-meaning pair is presented, the new agent updates its connection weights according to its weight-update rule, W, and the whole process is repeated.

Note the use of the term *environment*. Smith makes an important distinction between this and the entire space of possible meanings. As previously mentioned, agents in Brighton's simulations are potentially exposed during learning to any meaning from the entire meaning space, but Smith argues that it is not necessarily the case that meanings agents will experience in the real world will encompass the entire space of possible meanings in this way. There might be meanings, that whilst logically possible, are not likely to ever be experienced in a child's learning environment (for example, whilst one might expect to hear of a man driving a car, one wouldn't expect to hear of a car driving a man). Rather than manipulate the structure of the entire meaning space, as Brighton does, Smith's experiments focus on the agent's environment, which is taken to be a subset of the space of possible meanings. He defines two basic types of environment: unstructured, in which items are drawn at random from the space of possible meanings; and structured, in which they are taken from a contiguous area of this space. In addition to manipulating the *structure* of the environment, Smith also looks at environment *density*, that is the proportion of the total space of possible meanings that it contains.

The thrust of his experiment is to explore the emergence (of lack of) of compositional grammars using six different environment structures (3 different structured environments and three different unstructured environments, of sparse, medium and high density) under conditions where there is no bottleneck on transmission (i.e. where agents are exposed to all meanings from that environment during the learning phase) and for various different bottleneck sizes.

Under conditions where there is no bottleneck, he found that the emergence

of compositionality was infrequent; however, when it did occur, it was more likely to be in sparse, (and occasionally medium-density) structured environments. In all high-density environments, and all unstructured environments, the emergence of compositionality was extremely unlikely. The fact that the emergence of compositionality is infrequent is explained by the fact that in the absence of a bottleneck, compositional languages and holistic languages are equally stable. Given that this is the case, and given that the initial random languages of the agents are holistic, it is perhaps surprising that compositionality *ever* emerges under these conditions. Smith's further analysis of his results shows that the system seems to be sensitive to the compositionality present in the initial random system – although it does not guarentee the emergence of a compositional grammar, a higher degree of compositionality at the outset, unsurprisingly, favours the development of full compositionality. So, perhaps the most important question, is why is this only the case for medium to low-density structured environments?

The answer is quite simply that in a structured environment, distinct meanings tend to have feature values in common, thus if in the initial random system, there is a tendency to express a given feature value with a particular substring, it is possible that this may spread to cover all meanings involving that feature value. In an unstructured environment, very few meanings are likely to share any given feature value, so the potential for this to happen is much reduced. Furthermore, in a *high-density environment*, even if it is structured, and there happen to be a number of meanings that share a common substring for a common feature value, due to the large number of meanings containing that feature value, there are likely to be a lot more that do not share it, thus the chances of that substring spreading cover all of them is not very high. Thus compositionality is most common in low- to medium-density environments that are highly structured.

Once a bottleneck on the transmission of the language is introduced, the emergence of compositionality becomes almost inevitable. Almost all the

languages resulting from the simulations show some degree of compositionality, and highly compositional languages emerge with high frequency. Significantly, they emerge most frequently when the environment is structured. The size of the bottleneck also has an important influence. Whilst for structured environments, the emergence of highly compositional languages is almost guarenteed for *any* bottleneck, in unstructured environments this is not the case. For very tight bottlenecks, although compositionality is still favoured, languages tend to be only *partially* compositional, not fully so. What constitutes a *very tight* bottleneck depends primarily on the density of the environment: in very dense environments, both structured and unstructured, the emergence of highly compositional grammars is seen when agents are able to observe 40% or more of the environment during learning. However, when this figure is dropped to only 25%, the degree of compositionality seen in languages emerging in unstructured environments starts to drop. For medium-density environments, the figure at which this transition is seen is slightly higher – when agents are able to observe 60% or more of the environment during learning, highly compositional languages are seen in both structured and unstructured environment. However, at 40%, once again, the degree of compositionality seen in languages emerging from unstructured environments starts to drop – even more so than is the case for high-density environments. In sparse enironments, even when agents are able to observe *80%* of the environment during learning, the languages emerging when that environment is unstructured are generally only partially compositional. These observations can in part be explained by the fact that in structured environments, meanings are likely to share feature values with several other meanings, leading to the maximum advantage for compositionality especially where the bottleneck is tight. Thus we would expect to see highly compositional languages emerging across a whole range of bottleneck sizes. In an unstructured environment however, it might be entirely possible for a meaning to have a value for a particular feature that is not shared by any other meaning observed by the learner – in which case compositionality

would provide no advantage whatsoever. This is clearly more likely to be the case when the bottleneck is tight and thus a very small proportion of the environment is ever observed by any one agent.

Thus Smith concludes that the presence of bottleneck is crucial to the emergence of compositionality. In the absence of a bottleneck, highly compositional languages are unlikely to evolve, and only do so (with very low frequency) when the environment is structured and at least moderately sparse, due to the increased potential for compositionality to spread arising from the sharing feature values between meanings. However, in the presence of a bottleneck on cultural transmission compositional languages reliably emerge from the random initial holistic mappings of the agents. Only for very tight bottleneck values does this break down.

Willem Zuidema [93] has done his own investigations regarding the process of cultural transmission and used his results to put a new spin on Gold's learnability result [35], dicussed in Section 2.2. Gold concluded that in the absence of negative feedback, infinite languages are effectively unlearnable without contraints on the search space, leading Chomsky to postulate the existence of a Universal Grammar [19]. Using a simple learning algorithm inspired by Kirby's described above, Zuidema examines the ability of single agents to acquire a range of context free languages. In accordance with Gold's prediction, he found that whilst some languages were indeed learnable, others were not. He presents an example whereby agents are presented with a set of three strings from a recursive target language, and shows three different grammars that might have produced these strings, each of which an agent may exhibit during the process of grammar induction. However, agents always acquire the most general of these grammars. More restrictive languages, in which some of the strings from the most general case are not valid, are unlearnable. He extends this result to conclude that there are many target grammars that could never be correctly learnt, no matter how many sentence presentations are made. However, there are *some* that are always

learnt successfully, and yet others that can be learnt correctly but only some of the time. Thus, when the agents are incorporated into a generational framework, where each agent learns its grammar from a set of sample sentences generated by the previous one, he observes that the languages being employed by the agents tend to drift towards these more learnable languages as the simulations progress. Thus he concludes that there *is* indeed a constraint on the possible targets that a learner might have to acquire, but that this is not internally imposed by innate knowledge about the structure of language, as Chomsky has concluded. Instead, it is externally imposed by the process of cultural transmission: the languages that children must learn during the process of language acquisition will only ever be languages that other children have already successfully learnt before them.

Another limitation of the Iterated Learning Model as described so far is the simplicity of the population model employed. Smith and Hurford argue that this is a serious weakness [71] due to the apparent importance of factors such as population structure and demography in language evolution in the real world. Thus they have extended the model to the case where populations consist of multiple individuals. Instead of only having two agents in the simulation at any given time, one learner and one speaker from whom the learner receives utterances, these simulations involve a population of n agents, and a single learner. Each learner has p cultural parents drawn from that population, meaning that it only receives utterances from a subset of the total population. When an utterance is required, agents are drawn at random from the pool of cultural parents, with replacment. At the end of each generation, one speaker is removed from the population, again at random, the current learner becomes a speaker, and a new learner enters the simulation.

The first interesting finding that Smith and Hurford encountered due to this simple change in the model was a rapid increase in the length of the right hand sides of rules over generations, due to the addition of strings of meaningless terminal characters. They attribute this behaviour to the fact

that the learner will be sampling utterances from several speakers, resulting in exposure to multiple overlapping but non-identical grammars and consequent inconsistent training data, in conjunction with the greedy nature of the induction algorithm employed by the model. This, they claim results in overgeneralisations of the type that introduce the "spare" characters, and because agents do not consider multiple possible grammars at one time, nor are they capable of backtracking, they are unable to recover from this. In order to overcome this problem, a number of suggestions are made for ways in which the greedy nature of the algorithm could be reduced, but in the end, a simpler approach is opted for: simply fostering in the agents a bias towards signal simplicity, which means that when an agent comes to produce an utterance, it will favour the rule in its grammar that has the shortest right hand side. Thus over-generalisations that introduce additional terminal characters into the rules will still be made, but these rules will not be propagated. With this additional bias in place, they find that the results of Kirby's initial simulation [44] are indeed generalisable to larger population sizes: using a population size where $n = 10$ and varying the value of p such that $1 \leq p \leq 10$, compositional grammars are seen to emerge for the majority of runs. However, there appears to be an optimal value of p at around $p = 4$ or 5, where the grammars emerging are superior in terms of their size, coverage of the meaning space, and communicative accuracy. The number of cultural parents also appears to affect the *speed* of convergence, with the most rapid convergence on a shared compositional grammar also occuring for values of p where $p = 4$ or 5. Thus Smith and Hurford conclude that the previous model can indeed be extended to larger populations, and also that the number of cultural parents an agent has does have an impact on the structure of emergent languages and the speed with which they evolve. However, they stress the need to develop the model further to encompass situations where there is gradual population replacement and where learning interactions are unrestricted (i.e. where learners may speak to each other) as opposed to the strictly generational turnover approach applied to date.

Vogt has addressed this and other issues in his studies which attempt to unite Steels' grounded approach to lexicon emergence (using the THSim toolkit [84]) with Kirby et al's multigenerational iterated learning model, with interesting results [85]. Agents in his simulation play language games similar to those described in Section 3.1 where they observe objects in their environment and attempt to communicate to each other about them. The environment in question is composed of a set of geometrical coloured objects of different shapes. Agents attempt to form distinctions between these objects based on four perceptible features: the red, green and blue components of the **rgb** colour space, and shape, which is determined by the ratio of the area of the object and the area of the smallest bounding box that can be drawn around it. Agents create distinctions between these objects in a manner similar those in [75] and then engage in one of two different language games:

- **The observational game** in which the speaker picks an object from the environment, and indicates to the hearer which object it has picked. The speaker then produces an utterance for that object, and the hearer attempts to interpret it. If the speaker manages to decode the utterance to the correct meaning, the game is considered successful.

- **The guessing game** in which the speaker picks an object from the environment, but *does not* indicate to the hearer which object it has picked. The speaker then produces an utterance for that object, and the hearer attempts to interpret it. Once it has decoded the utterance, the hearer makes a guess as to which object the speaker was referring to. If it guesses correctly, the game is considered successful.

The greatest difference between this study and those of Steels [72, 73, 75] is the incorporation of these games into a transgenerational model. The iterated learning model employed is in most respects very similar to that used

by Kirby [43, 46]: at any given time there are two agents in the population, one learner, one speaker. Learners enter the population with no knowledge of the language at all, and also no knowledge of the objects in the environment, or how to distinguish them from each other. By engaging in interactions with the speaker, they attempt to acquire this information. After a predefined number of interactions, the speaker is removed from the simulation, the learner becomes a new speaker, and a new blank agent is introduced to the system as the new learner. Thus, vertical transmission of language from speaker to learner across a multiple generations is achieved, in the same way as described in Section 3.3.

However, the grammar induction algorithm employed here, although broadly to similar to Kirby's, does differ from it in some significant ways. Firstly, as well as storing a set of grammatical rules that the learner has induced from its exposure to the speaker's language, Vogt's agents also store a list of *instances*: for each game where the learner receives an utterance and sucessfully identifies the topic, the string presented plus its meaning are added to this repository. A count of the frequency with which each string-meaning pair has been observed is also stored. It is these stored instances themselves rather than any grammar rules that have already been acquired that are used as the substrate for the induction process. The justification for this is that human learners do seem to store both whole utterances as exemplars and generalizations of these. This has the advantage the learners are not forced to abide by previously made generalizations which allows them some ability to backtrack if those generalizations prove incorrect.

Secondly, novel utterances are not automatically added to the grammar as holistic rules in the way that they are in Kirby's simulations. Instead, the learner seeks to make generalisations from those utterances and only resorts to adding holistic strings if it is unable to do so. Agents will first seek to exploit any previous rules that have been learnt, where, for example, the agent is already capable of decoding part of the sentence. If this is the case,

new grammar rules will be created to cover the remainder. If the agent is not able to decode the sentence at all, the utterance is compared to each of the items in the repository of stored instances to see if the chunking operator might be applied.

Thirdly, the exact nature of the chunking operator is also slightly different to that described by Kirby: recall that on page 63, we stated that in Kirby's simulations rules are compared on a pairwise basis and if any two rules are found to differ by just a single part of their meaning, and also the strings associated with those meanings are found to differ by a single substring, then the difference in the meanings is attributed to the differences in the strings. Both rules are removed from the grammar, and new rules are introduced to reflect this. Vogt's chunking operation however, is based on based on van Zaanen's Alignment Based Learner [83], and thus, rather than searching for differences between strings, it looks for *alignments*: pairs of utterances whose strings share a common substring, and whose meanings also share common values. In a sense, Kirby's operator can also be said to be looking for alignments. Looking at the following example, where the agent has rules in its grammar stating that the string *j,o,h,n,l,o,v,e,s,m,a,r,y* means and *loves(john, mary)* and the string *j,o,h,n,k,n,o,w,s,m,a,r,y* means *knows(john, mary)* we can see that this can be viewed either as a difference between the predicate element of the meaning (*loves(x, y)* vs *knows(x, y)*) and the two substrings *l,o,v,e* and *k,n,o,w*, or an alignment between the subject and object elements of the meaning (*p(john, mary)*) and the substrings *j,o,h,n....m,a,r,y*. However, in Vogt's simulations, alignments can only be made *either* at the start of the sentence *or* at the end, and thus generalization such as that in the example, where a new rule is created from the middle portion of the string, would not be made. This helps to prevent the occurence of meaningless characters in sentences and the explosion in string length seen by Smith [71]. Furthermore Vogt's chunking operator does not require the meanings of the utterances to be aligned on *all but one* of their features in the way that Kirby's does. This

results in a much wider range of possible generalizations.

Fourthly, Vogt's chunking algorithm does not make all possible generalizations in a single pass as described in Kirby's algorithm. When a new utterance is encountered that cannot be added to the existing grammar by exploiting the rules that have already been induced, the agent searches through the repository of stored instances looking for possible alignments of the type described above. When more than one possible alignment is found, that containing the substring that has been observed most frequently is chosen. If there is tie, the largest alignment is used.

Finally, rules that have been learnt are never *deleted* from the grammars of Vogt's agents. Instead he uses a series of sophisticated update rules to ensure that those which have been used most successfully will be favoured. This again means that agents are not committed to previous generalisations they have made, and gives them the ability to backtrack.

Using the experimental conditions described above, Vogt first examines the types of grammars that emerge in the absence of a bottleneck (i.e. where agents are exposed to the entire meaning space during learning). He finds that even in the absence of a bottleneck, compositional behaviour does reliably emerge. This is somewhat at odds with Smith's result [70] described above, in which the emergence of compositionality in the absence of a bottleneck was a rare occurence. This is perhaps because Vogt's induction algorithm is more strongly biased towards relating parts of the meaning to parts of the string than Smith's networks. However, what is notable about Vogt's results is the differences in outcome between simulations where agents were playing the *observation game* and those where they were playing the *guessing game*. In both cases, compositional behaviour emerges rapidly. In both cases, there are seen to be random fluctuations in the amount of compositionality observed, where languages will change from compositional to holistic ones. However, compositionality is always ultimately restored. What is interest-

ing is that a) although full compositionality takes longer to be achieved in the guessing game than the observation game, the degree of compositionality seen in the latter is generally higher, b) the guessing game seems less prone to sudden losses of compositionality than the observation game, but it also seems to be more sensitive to its effects; in the observation game, when languages become holistic, agents seem still able to communicate accurately, whilst in the guessing game they cannot, and c) the amount of *coherence* seen in the guessing game (i.e. the degree to which agents give the same name to the same object) is much greater than in the observation game.

Vogt then goes on to see what happens when additional agents are added to the population. Instead of a single learner and a single speaker, he increases the number of agents to *three* of each. This appears to have quite a catastrophic effect. As with the smaller population size, for both the observation game and the guessing game, the degree of compositionality observed quickly rises to a high level. This time however, it does not appear to be as stable, and soon starts to decrease again. The grammars emerging from the guessing game do seem to be slightly more resilient than those emerging from the observation game however, and in some cases compositionality persists, or even recovers for a while. The effect of this loss of compositionality on communcative accuracy is the same as before: for the observation game, the transition to holistic grammars does not prevent agents from communicating effectly but for the guessing game, lost of compositionality results in a loss of accuracy. Coherence in both cases is seen to increase as compositionality decreases, and is again higher for the guessing game than for the observation game.

Finally Vogt repeats this experiment but introduces a bottleneck on the transmission of language: language games that the agents engage in during the learning phase now only cover 50% of the objects in the environment. When the population size is only two (one learner, one speaker), this has the striking effect of causing compositionality to emerge quickly to high level,

and to remain at a high level throughout the duration of the simulations. This, of course, is perfectly in keeping with results from Smith [70] and Brighton [12, 11, 10] which suggest that the imposition of a bottleneck is highly conducive to the emergence of compositional languages. When the size of the population is 6 (3 learners and 3 speakers) again a high and stable level of compositionality is seen – but only for the guessing game. In the observation game, compostionality tended to persevere for longer than in the absence of a bottleneck, but it still ultimately disappears and simulations revert to holistic languages.

3.4 Computational Models of Case and Word Order

To date, not a lot of work has been done regarding the modelling of case and word order and the interrelationship between them. However, there are two studies of significance that should be discussed here.

Firstly is that of Luc Steels, who has attempted to extended his grounded paradigm whereby agents engage in "lanugage games" regarding real-life objects in the world around them to negotiate a shared language. In his study of 2002 [79] he describes experiments where agents observe dynamic scenes played out before them. They categorise these scenes into a series of events and micro events, and the agent acting as speaker in a given interaction will select a single one of these scenes from its recent memory and use items from a predetermined lexicon to create a unique utterance with which to describe it. The role of the hearer is to decode the utterance and to identify which of the recent events the speaker is referring to. If it succeeds in doing this correctly, the game is considered successful. Based on the assumption that speakers of a language wish to maximise their communicative accuracy,

agents are primed to try and reduce ambiguity wherever possible. However, they are not compelled to make explicit all elements of the scene they are described. For example, Steels uses the scenario where a red ball moves away from a smooth green ball. Also present in the environment are a hand and a box, although they do not take part in the event in question. The speaker simply describes this event as *SMOOTH MOVE-AWAY-FROM*. It is not necessary for it to specify that the smooth object is also a green ball, as this information is clear from the context; similarly it does not need to specify which second object is involved in the event because this is the only *MOVE-AWAY-FROM* event in recent memory that involves the smooth green ball. Finally, it is not necessary for it to make explicit in this case whether the smooth green ball is *doing* the moving away from, or *being* moved away from, again because there is no other *MOVE-AWAY-FROM* event in recent memory that involves the smooth green ball, so this level of disambiguation is not required.

Thus Steels found that even in this very simple case a relatively good degree of communicative success can be achieved simply by virtue of the fact that agents have a shared context. The next stage of his experiment, however, was to try and prompt agents to introduce markers for semantic relations to make explictit the object-event relationships in situations where it is necessary to distinguish between one event and its exact opposite. In order for the speaker to know when such additional markings would be helpful, the technique of "re-entrance" is employed. When the speaker produces an utterance, it uses its own grammatical knowledge and memory of recent events to interpret the utterance and attempt to ascertain whether it would be ambiguous to the hearer as to which event is being described. If it finds this to be the case, introduces a marker to specify what the role is of one of the objects in the event. For example, it may change the utterance describe above from *SMOOTH MOVE-AWAY-FROM* to *PO SMOOTH MOVE-AWAY-FROM* where the morpheme *PO* indicates that the smooth green ball is object doing the mov-

ing in the *MOVE-AWAY-FROM* event. If the hearer is already familiar with the marker in question, then the utterance can be interpreted unambiguously and communication has been successful. If however, the marker is new, then the hearer must make a guess at its meaning which can be confirmed through further usage.

In this way, agents are able to make explicit the semantic relations between objects in the events being described. However, the "language" emerging still does not look much like the case systems used in natural languages: there are large numbers of different semantic markers, unique to each event type within the environment. In order to address this, Steels has performed a final experiment in which agents attempt to generalise across semantic markers by use of analogy. When disambiguation is necessary, instead of simply introducing a new marker, agents first try to see whether there is already a marker that expresses an analogous event-object relationship that can be exploited. The process by which Steels' agents determine analogy is interesting: it is basically a mapping from a source event for which markers already exist to a target event for which semantic markers do not yet exist. Agents decompose the events into primitive microevents. So, for example, *WALK-TO* event involving *WALK-TO-1*, that is the agent doing the walking, and *WALK-TO-2*, that is the target being walked towards, might be broken down into the following sequence of microevents: the agent does not move, the target does not move, the agent approaches the target, the agent touches the target. Similarly, a *MOVE-INSIDE* event that also involves two objects, *MOVE-INSIDE-1*, the agent moving and *MOVE-INSIDE-2*, the location into which the agent is moving, might be broken down into these microevents: the agent is visible, the location is visible, the agent does not move, the location does not move, the agent approaches the target, the agent touches the location, the agent becomes invisible. If all the microevents in the target event can be mapped onto microevents involving the same object in the source event, then the two events are considered analogous and any markers

associated with the source event will be used for the target. Thus in the example given above, *MOVE-INSIDE* is the source event, and *WALK-TO* is the target, then analogy exists, as the four microevents into which *WALK-TO* can be decomposed all exist in the *MOVE-INSIDE* event as well. However, were *WALK-TO* the source, and *MOVE-INSIDE* the target, analogy would not be found.

The result of this exploitation of analogy in order to re-use markers is that agents end up using a far smaller number of different markers, which will be shared between different events with different semantic commonalities. In short agents have constructed their own semantic categories. Steels claims that analogy leading to the re-use of existing forms to provide new meanings is a fundamental driving force in the introduction of new layers of grammar, and proproses that the same principles could be put to use in evolving grammars for tense, determination, sentence structure etc.

The second computational study of case that I am about to describe is strictly speaking a study of case and word order *acquisition*, rather than their *emergence*, but the conclusions that can be drawn from it have important implications for the emergence of such behaviours, so it is worthy of discussion here.

The study in question is by Lupyan and Christiansen [53, 52], who use simple recurrent neural networks to model the acquistion of a series of artificial languages, which exemplify each of the six possible orderings of subject, object and verb, plus a freely ordered language. Each of these possibilities is used with and without case marking, giving a total of fourteen languages all together. After training on a subset of the language in question during which agents are presented with sentences from it in sequential form, and given information about whether each word in the sentence is the subject, direct object, indirect object, genitive noun or verb, agents are then tested on the full language. Lupyan and Christiansen found that for the seven

languages that included case markings, agents were successfully able to predict the parts of speech for 100% novel utterances, irrespective of the actual word order and whether it was fixed or free. However, the degree of success with languages which did not have case markings varied quite considerably. Near perfect performance was obtained for only two word orders – SVO and VSO. On SOV languages, performance was considerably poorer, and with the other three possible fixed word orders, poorer still. Unsurprisingly, free word order languages without case markings faired very badly indeed. This result is interesting because (according to Greenberg's Universal number 1 [38]) the vast majority of languages in the world are one of these three types: SVO, VSO or SOV. The other three orders, despite being seemingly equally good candidates are very much underrepresented. Furthermore, (according to Greenberg's Universal number 41 [38]), verb final lanugages almost always have a case system. Thus, the data obtained by Lupyan and Christiansen seems to reflect quite accurately the frequencies of different word orderings in the languages of the world.

This experiment was followed up by one to investigate the interrelationship between case and word order: using a similar paradigm to the first experiment, languages were created which exhibited different degrees of case-marking, from only genitive markers to marking on 100% of the nouns. This was intended to reflect the fact that in some natural languages, case markings are phonologically ambiguous, or there may be certain nouns which do not take case markings at all. Five such languages were created. In addition, five different conditions regarding the strictness of word order were introduced, modelled on the word orders permissable in real-world natural languages. These ranged from a language based on English with fully fixed word order (SVO) to one with completely free word order. There were also three intermediates with increasing degrees of freedom based on Italian (predominantly SVO, but all other combinations are possible where they can be disambiguated from other cues [4]), Polish (predominantly favours orderings

in which the subject preceeds the object [41]) and Turkish (almost completely free word order – approximately 50% of utterances are SOV, with SVO, OVS and OSV being the next most common [69]). The combination of the five different degrees of case-marking with the five different possibilities for strictness of word order resulted in 25 test languages, in which agents were trained and tested in the same manner as in previous the previous experiment. As expected, the results showed that the more consistent the word order of a language, the less the performance of agents learning that language could be improved by the addition of case markings.

Finally Lupyan and Christiansen carried out some experiments whereby they compared the performance of their networks to that of children learning real languages. Slobin and Bever [69] have shown that children acquiring languages such as Serbo-Croatian (which use a mixed strategy for disambiguating semantic relations in which neither case nor word order can be fully relied upon, meaning that it is necessary to attend to both) are slower in learning to reliably determine event-object relationships accurately than children acquiring languages which either have completely fixed word order, such as English, or reliably expressed case markings, such as Turkish. Lupyan and Christiansen found that performance of their networks on the artificial languages created to model the properties of these languages were very similar: languages in which case and word order are both required in order to determine semantic relationships are harder to acquire.

Thus what both Steels' and Lupyan and Christiansen's studies appear to show is that where disambiguation is necessary in order to determine event-object relationships, the use of case markings will aid this process greatly.

3.5 Summary

It is the aim in the current study to see whether the Iterated Learning Model type approach can be extended to generate the emergence of case-like behaviour by introducing such a need for disambiguation. So far, one of the key features of grammars that emerge from simulations based on models such as Kirby's is the use of word order to specify meaning distinctions. However, it is worth noting that in many natural languages, meaning distinctions are not wholly specified by word order – even in English, some freedom of word order is allowed, and other languages allow much more. This is generally accompanied by a much richer case system than that found in English. Thus it is hoped that it will be possible to exploit this relationship and stimulate the emergence of case-like behaviour by the introduction of a degree of word order flexibility into the system. In the chapter that follows, I shall go on to describe in detail my implementation of the Iterated Learning Model, and in Chapters 5 and 6 the effect of introducing some freedom of word order into this model will be described.

Chapter 4

Implementing an Iterated Learning Model

As discussed in Chapter 3, the aim of the work presented in this thesis is to investigate whether the Iterated Learning Model developed by Kirby et al [44] can be used to model the emergence of further features of natural language, in particular the use of case-markings to denote semantic relations. To this end, an implementation of such a model is presented. This is based on the description given in Kirby [44].

4.1 Basic Features of the Model

Individuals in the system are equipped with an induction algorithm (for learning grammars), a parsing and production algorithm (for interpreting incoming utterances, and producing new ones) and an invention algorithm (for dealing with meanings that they are otherwise unable to express). All agents are identical and start life with no grammar at all. An overview of

their lifecycle is shown in Figure 4.1.

At any given time, there are two agents in the system, a speaker and a learner. The speaker produces utterances by consulting the grammar it has learnt during its period as a learner, and producing a string of characters for the meaning it wishes to express, or if unable to produce a string in this way, by resorting to invention. In the case of the very first learner, it will be necessary to resort to invention for every utterance it produces, as it has not had the opportunity to learn a grammar from a previous speaker. This means that the initial language will be entirely holistic. The string that results from either production or invention is then passed to the learner along with the intended meaning. The learner first attempts to parse the string, and if able to do so takes no further action. It does not require the meaning that *results* from this parse to be the same as that which the speaker intended, so long as the string can be parsed to give *some* meaning. This is important because it helps to ensure a one-to-one mapping between meanings and strings, the significance of which will be discussed in Section 4.2.1. If the string cannot be parsed at all, however, the learner incorporates it into its grammar in conjunction with the correct meaning. This process is repeated a fixed number of times – in the simulations described here, 100. The speaker is then removed from the simulation, the learner becomes the new speaker, and a new learner with no grammatical knowledge is "born". At this point, the grammar rules that the new speaker has aquired during its period as a learner are "shuffled". This helps to ensure that the speaker's choice of grammatical rule is not biased towards those which were acquired first. Simulations are typically run for 5000 generations.

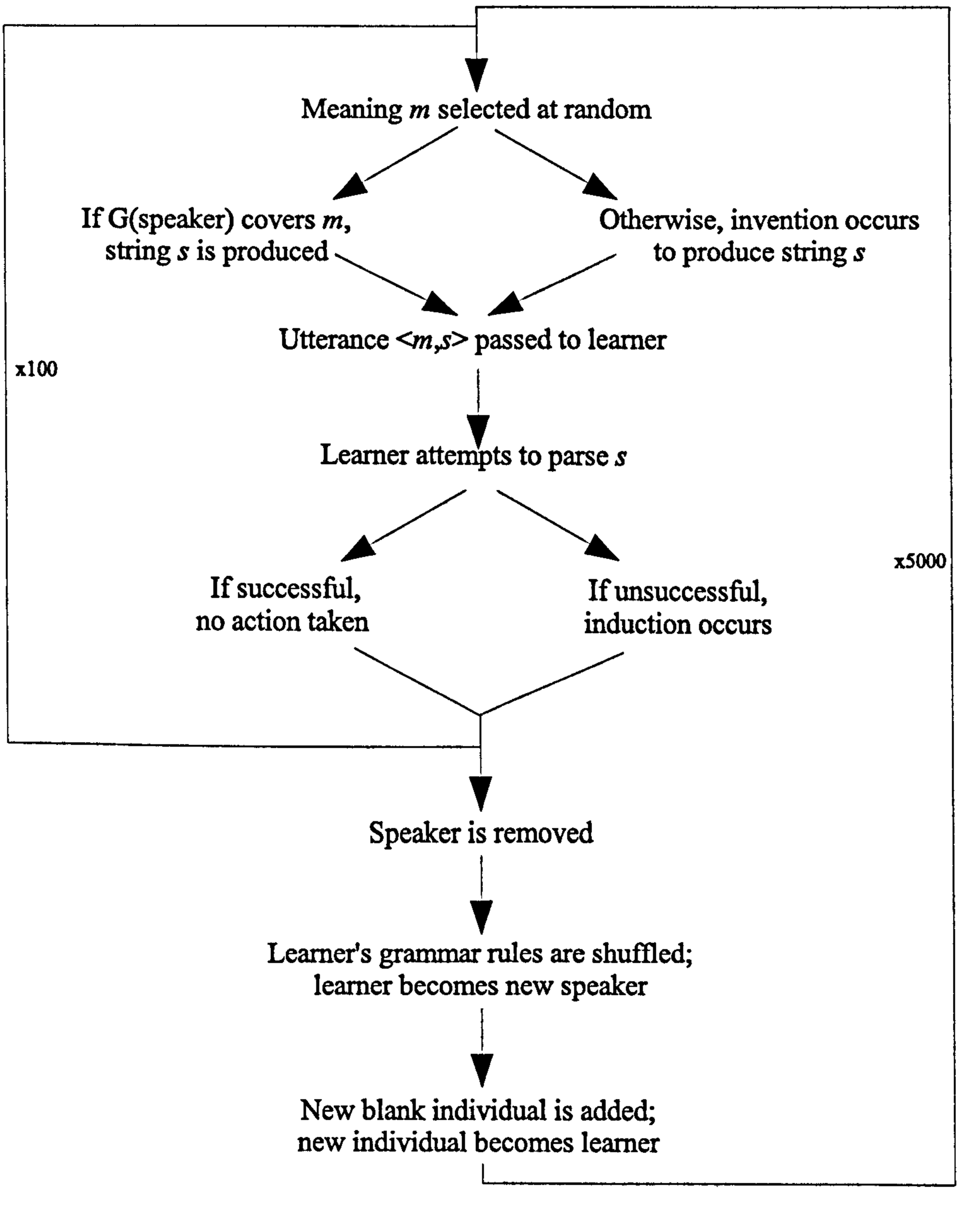

Figure 4.1: *The life cycle of the simulation.*

4.1.1 The meaning space

Meanings to be expressed by the speaker are drawn at random, with replacement, from a simple meaning space. These meanings are viewed as having been provided by external world, perhaps something particularly salient to the speaker agent, prompting it to try and communicate some information to the learner. The space is composed of a number "who did what to whom" type propositions, taken from a set of actions or events that might occur, such as "loves", "hates", "hits" etc. and a set of individuals that might participate in those actions or events, such as "john", "mary", "kate" etc. Elements from the two sets are combined, with members of the former acting as predicates, and the latter set their arguments, to give statements of propositional logic such as *loves(john,mary)* written (for ease of implementation) in the vector format [loves,john,mary]. In each vector, the first position represents the action or event being described, the second the "actor" in that event, and the third the party being "acted-upon". As in Kirby [43], the constraint has been included that "actor" and "acted-upon" must be distinct: meanings such as [loves,john,john] are disallowed. The "actor" and "acted-upon" elements of the semantics could be viewed as "subject" and "object" of the event being described; however, I have chosen to avoid these terms as they have syntactic connotations and it is an important feature of this model that agents have no syntactic knowledge. They are aware who is the "actor" and who is the "acted-upon" in the event being described because they have presumably witnessed it happening, and not because they have any innate knowledge of subjects, objects or any other syntactic category.

In the simulations described below, the meaning space consists of 5 individuals, plus 5 predicates resulting in a total of $5 \times 5 \times 4 = 100$ possible meanings. Unlike Kirby [44], no embedded predicates are used here; the meaning space contains only simple propositions as in his earlier studies.

Given that the size of the meaning space is 100 (from which instances are chosen randomly, with replacement), and that each agent hears only 100 utterances during its period as a learner, the chance of a given agent being exposed to *all* possible meanings is extremely small. This is crucial to the emergence of natural language-like behaviour, as discussed in Chapter 3, as it imposes a "bottleneck" on the transmission of information.

4.1.2 The grammar and the parser

The grammar itself represents what the agent *knows* about the language of its speech community, and is basically a list of rules for how to construct sentences. Each agent starts life with a completely empty grammar, and adds rules to it on the basis of utterances it has heard during its period as a learner. The representation used is a context free grammar enriched with simple semantics, again as described in Kirby [44]: non-terminals have a single argument attached to them which conveys semantic information. Thus a rule such as

$$\text{NT/john} \longrightarrow \text{j,o,h,n}$$

indicates that the string "j,o,h,n" is a word of category *NT* meaning *john*: the left hand side of the rule is made up of the syntactic category of the string, plus its meaning, and the right hand side consists of a string of either terminal categories as in the above example, or non-terminal categories and their associated semantic labels, such as the following:

$$\text{s/[P,X,Y]} \longrightarrow \text{NT1/P, NT2/X, NT3/Y}$$

The parser is a straight-forward top-down deterministic parser, augmented to cope with the semantic representations associated with the rules in the grammar. Its role is to parse incoming utterances, thereby identifying whether

they are covered by an agent's current grammar. The same algorithm is also used in the production of outgoing utterances. It operates by choosing a rule from the grammar, taking each of the characters on the right hand side of that rule in turn, and expanding those of them that are non-terminals If any of the characters in the string returned are non-terminals, it will expand those too. Non-terminals are expanded by finding the first rule in the grammar with that category on its left hand side and whose semantics can be unified with those required and repeating the process on that rule.

The deterministic nature of the parser is very important: if the grammar allows more than one way of expressing a given meaning, only one will ever be used, because the agent simply moves down its grammar searching through the grammar rules in the order in which they appear. For this reason, rules are re-ordered randomly during the transition from learner to speaker, thus ensuring that those learnt first do not necessarily appear highest in the grammar. The fact that the parsing algorithm is deterministic is another feature of the simulation which helps to ensure a one-to-one mapping between strings and meanings, which, as Smith [70] has shown, is crucial to the emergence of language in simulations such as these. This point will be returned to in Section 4.2.1.

4.1.3 The induction algorithm

The grammar induction algorithm consists of two basic phases, as in Kirby [44]. The first is a simple incorporation step, which involves building the simplest rule which will relate the string heard to its intended meaning. A the new rule has the non-terminal symbol s on its left hand side, and the semantic argument associated with it is simply the meaning vector for the utterance concerned. The right hand side is the string part of the utterance.

Thus if the utterance had been

< [loves,john,mary] j,o,h,n,l,o,v,e,s,m,a,r,y >

the rule

s/[loves,john,mary] ⟶ j,o,h,n,l,o,v,e,s,m,a,r,y

would be created.

The second phase of the induction algorithm is to make generalisations between this new rule and others already present in the grammar. This involves comparing rules on a pairwise basis and seeking to create a new rule that will subsume them both. To this end, there are two basic operations available to the agent:

- If rules A and B differ only by non-terminals X and Y, and if changing Y to X would make them identical, then rule B is removed, and all other instances of Y in the grammar are changed to X.

- If the semantics of rules A and B differ by the value of a single element whose meanings are a and b, and their strings differ by substrings α and β, a and b are replaced by a variable x, and α and β are replaced by a non-terminal, N whose meaning is x. New production rules are created from N to strings α and β with meanings a and b respectively.

In the simulations described, these two operations are actually served by a total of *four* basic heuristics. These are ***findchunks***, ***findchunk***, ***mergeable*** and ***subrule***, whose implementations are outlined below. Whenever a new rule is added to the grammar, it is compared with each of the existing rules, and the heuristics are applied in the order **subrule**, **mergeable**, **findchunks**, **findchunk**. For each new rule that is created by the application of

these operations, the process is repeated until no further simplifications to the grammar are possible. There is also a function that removes any duplicate rules that may have been created during the induction process from the grammar.

Findchunks

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_1/m_2 \longrightarrow \sigma_2$
  if m_1 and m_2 differ only by values v_1 and v_2 respectively
  and σ_1 and σ_2 differ only by substrings λ_1 and λ_2 respectively
    add new rules $\mathcal{N}_2/v_1 \longrightarrow \lambda_1$ and $\mathcal{N}_2/v_2 \longrightarrow \lambda_2$
      where $\mathcal{N}_2 \in \mathcal{N}$
    replace r_1, r_2 with new rule $\mathcal{N}_1/m_3 \longrightarrow \sigma_5$
      where $m_3 = m_1$ with v_1 replaced by variable $\mathcal{V}$
      and $\sigma_5 = \sigma_1$ with λ_1 replaced by $\mathcal{N}_2/\mathcal{V}$

This heuristic identifies differences in meaning between two rules, and attributes them to differences in the strings representing those meanings. Thus if the two rules being compared are

s/[loves,john,mary] $\longrightarrow$ j,o,h,n,l,o,v,e,s,m,a,r,y
s/[loves,john,kate] $\longrightarrow$ j,o,h,n,l,o,v,e,s,k,a,t,e

this heuristic would notice that the only difference in the meanings of the two strings is the individual in the position in the meaning vector representing the person being acted upon: "mary" in the first case and "kate" in the second. It would also note that the two strings are almost identical apart from the final substring, *m,a,r,y* in the first rule and *k,a,t,e* in the second. It

will thus conclude that the substring *m,a,r,y* means "mary" and that *k,a,t,e* means "kate", and alter the rules of the grammar to reflect this, by replacing the two original rules with the following:

s/[loves,john,X] ⟶ j,o,h,n,l,o,v,e,s,NT/X
NT/mary ⟶ m,a,r,y
NT/kate ⟶ k,a,t,e

Findchunk

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_1/m_2 \longrightarrow \sigma_2$
if m_1 and m_2 differ only by value v and variable $\mathcal{V}$ respectively
and σ_1 and σ_2 differ only by substring λ and non-terminal $\mathcal{N}_2/\mathcal{V}$ respectively
add new rule $\mathcal{N}_2/v \longrightarrow \lambda$
where $\mathcal{N}_2 \in \mathcal{N}$
replace r_1 with new rule $\mathcal{N}_1/m_2 \longrightarrow \sigma_2$[1]

This heuristic is very similar to the *findchunks* heuristic described above, except that when searching for differences in the meanings of the two rules being compared, rather than looking for two atomic values, an atomic value in one rule and a variable in the other are required. Similarly, the differences in the strings representing those meanings should be a string of terminal characters in the first case and a non-terminal associated with the variable identified in the other. For example, given the rules

[1]Clearly this new rule is identical to r_2. However, in the current implementation, r_1 is replaced with a duplicate rule rather than simply removed in order to facilitate the changes that will be made in Chapter 5.

s/[loves,john,jane] ⟶ j,o,h,n,l,o,v,e,s,j,a,n,e
s/[loves,john,X] ⟶ j,o,h,n,l,o,v,e,s,NT/X

it can be seen that they again differ semantically in terms of the value in the third position of the meaning vector – that representing the individual being acted upon. In the first rule, it is "jane" and in the second it is the variable X. The strings associated with these meanings again differ in their final substrings: the first rule ends with the string *j,a,n,e* whilst the second ends with an instance of the non-terminal NT. The *findchunk* heuristic would conclude from this that the substring *j,a,n,e* is an instance of the non-terminal NT and create a rule to this effect. Thus the first rule would be removed from the grammar and the following added:

NT/jane ⟶ j,a,n,e

Subrule

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_2/m_2 \longrightarrow \sigma_2$
  if σ_2 is a proper substring of σ_1
  and m_2 appears in m_1
    replace r_1 with new rule $\mathcal{N}_1/m_3 \longrightarrow \sigma_3$
      where $m_3 = m_1$ with m_2 replaced by variable $\mathcal{V}$
      and $\sigma_3 = \sigma_1$ with σ_2 replaced by $\mathcal{N}_2/\mathcal{V}$

The purpose of the *subrule* heuristic is to identify substrings in the newly incorporated utterances which can be attributed to one of the previously induced rules. For example, if the grammar contained the pair of rules

s/[loves,john,jane] ⟶ j,o,h,n,l,o,v,e,s,j,a,n,e
NT/john ⟶ j,o,h,n

then the subrule heuristic would identify that the semantic value associated with the second rule, "john", appears in the semantic vector associated with the first, and that the string on the right hand side of the second rule, *j,o,h,n*, is a substring of string associated with the first. Thus it would assume that this substring when it occurs in the first rule is an instance of the second rule, and make the necessary changes to the grammar. The first rule would be replaced with the following:

s/[loves,X,jane] ⟶ NT/X,l,o,v,e,s,j,a,n,e

Mergeable

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_2/m_2 \longrightarrow \sigma_2$
if m_1 and m_2 are unifiable and $\sigma_1 = \sigma_2$
replace all instances of $\mathcal{N}_2$ in the grammar with $\mathcal{N}_1$
OR
if $\mathcal{N}_1 = \mathcal{N}_2$ and m_1 and m_2 are unifiable
and σ_1 and σ_2 differ only by categories $\mathcal{N}_3/\mathcal{V}_1$ and $\mathcal{N}_4/\mathcal{V}_2$
and $\mathcal{V}_1$ and $\mathcal{V}_2$ occupy corresponding positions in m_1 and m_2
replace all instances of $\mathcal{N}_4$ in the grammar with $\mathcal{N}_3$

The final heuristic *mergeable* is responsible for identifying pairs of non-terminal categories which essentially serve the same syntactic function as each other, allowing the inducer to replace all instances of one category in the grammar with the other. Categories are judged syntactically equivalent

in this way if they occur in pairs of rules which are otherwise identical, such as

NT1/john ⟶ j,o,h,n
NT2/john ⟶ j,o,h,n

OR

s/[loves,john,X1] ⟶ j,o,h,n,l,o,v,e,s,NT1/X1
s/[loves,john,X2] ⟶ j,o,h,n,l,o,v,e,s,NT2/X2

In both these cases, the categories NT1 and NT2 would be selected for merging.

4.1.4 The invention algorithm

The invention algorithm is the process by which agents can produce utterances for meanings which are not covered by their grammars. This is very important in the initial stages of the simulation, for in the first generation, the speaker has no grammar at all: without some way of inventing utterances, language would therefore be completely unable to get off the ground. However, it continues to be important for many subsequent generations, due to the existence of the *learning bottleneck*: as previously mentioned, the total size of the meaning space is 100, from which items are drawn at random (with replacement), and speaker agents only make 100 utterances per generation. Therefore, it is extremely unlikely that any given learner will encounter every possible meaning. It is possible to calculate an estimate of exactly how many it will observe, using a measure called *coverage* [9], where the expected *coverage* c after R random observations drawn from a pool of N objects is as follows:

$$c = 1 - (1 - 1/N)^R$$

Therefore, in the case of 100 utterances per generation and a meaning space of 100, this figure would be

$$c = 1 - (1 - 1/100)^{100} = 0.6339$$

In other words, after 100 observations drawn at random from a meaning space of 100 items, we can expect to have seen 63.39% of the items in that meaning space. Thus there will probably continue to be meanings which are not covered by the agents' grammars for quite some time. Eventually, if compositionality starts to emerge, then agents will start being able to produce utterances for meanings they have not previously experienced and the number of meanings that cannot be expressed will decrease. If optimal compositionality is achieved, then agents should be able to express the entire meaning space even if there are meanings that they have not previously observed, and they will no longer be forced to resort to invention.

All this points to the fact that the inventor must be such that it preserves any grammatical structure that has already emerged, without adding any *new* structure. The algorithm used here is once again taken from that described in Kirby [44]. It searches for the closest meaning to the one required that the speaker *can* produce, deletes any string associated with the "incorrect" part of the meaning, and replaces it with a new substring.

Thus existing structure is preserved: if the agent is asked to produce a string for a meaning such as [hits,john,pete], and has rules in its grammar such as

s/[hits,john,X] ⟶ j,o,h,n,h,i,t,s,NT/X
NT/jane ⟶ j,a,n,e
NT/mary ⟶ m,a,r,y

then it will seek out the nearest possible meaning for which it can produce a string. This would be either [hits,john,jane] or [hits,john,mary]. Either

way, the part of the string associated with the *incorrect* part of the meaning, either "jane" or "john" would be deleted and replaced with a new substring, such as *b,l,i,p*, resulting in *j,o,h,n,h,i,t,s,b,l,i,p*, and the structure inherent in the grammar is preserved. The alphabet from which invented strings are drawn, and the maximum and minimum lengths of those strings, are supplied parametrically. In the studies describe here, the alphabet is simply the whole of the roman alphabet, and the string length is between one and three characters.

If however, the requested meaning had been [hits,pete,jane], then again the closest possible meaning would be [hits,john,jane]. However, in this case, the *incorrect* part of the meaning is "john" which is not expressed by any subpart of the string, but rather by the toplevel rule itself. Therefore there is no one substring that can be replaced in order to create the desired utterance, and a completely novel holistic string is used instead. Thus structure is not being introduced to the grammar that was not previously present.

Once invention has taken place, the speaker has essentially produced a string for the meaning that it was previously unable to convey. It will then add the string-meaning pair to its own grammar by using it as input to the induction algorithm. In this way, it ensures that the next time it is required to produce a string for that meaning, it will be able to do so.

4.2 A Compositional Grammar

As in the results presented in Kirby [44], the language spoken by the population of the simulation evolves over a number of generations from a simple vocabulary driven language, where each meaning is represented by an idiosyncratic string with no internal structure, to a fully compositional language, in which the meaning of the string is derived from the meaning of its parts

and the way they are assembled. In particular, separate syntactic categories for nouns and verbs emerge, which are combined in a fixed order which encodes meaning distinctions in a compositional manner. The percentage of the meaning space covered by the grammars is seen to increase from around 60% on average in early generations to 100% in later ones. This is accompanied by a dramatic decrease in the number of rules in the grammar from over 100 to begin with to very close to the minimal value of 11 (one "top level" rule, plus five rules for each of the five *individuals* and five for each of the five *actions*).

This can be exemplified by looking at sample grammars taken from agents existing at various points in a single simulation. Below is the grammar of the first agent in the simulation at the end of its life:

s/[loves, anna, mary] ⟶ c
s/[loves, anna, pete] ⟶ x
s/[sees, mary, anna] ⟶ t, b, s
s/[hates, anna, mary] ⟶ b
s/[adores, pete, kath] ⟶ b, p, o
s/[loves, anna, john] ⟶ k
s/[loves, kath, anna] ⟶ j, x, j
s/[adores, mary, kath] ⟶ z, g, h
s/[kisses, pete, anna] ⟶ c, v
s/[sees, kath, mary] ⟶ d, q
s/[adores, mary, anna] ⟶ m
s/[adores, john, kath] ⟶ f
s/[kisses, kath, john] ⟶ k, n, c
s/[sees, john, anna] ⟶ i
s/[adores, anna, kath] ⟶ h, n, s
s/[adores, john, pete] ⟶ s
s/[adores, john, mary] ⟶ j, f
s/[adores, mary, john] ⟶ i, f, o

s/[sees, john, kath] ⟶ r
s/[loves, pete, mary] ⟶ s, k, e
s/[adores, A, anna] ⟶ q, 1/A
1/kath ⟶ q, k
1/pete ⟶ j
s/[kisses, kath, A] ⟶ q, 1/A, g
s/[hates, mary, pete] ⟶ l, q
s/[loves, john, anna] ⟶ l, f, u
s/[sees, kath, john] ⟶ c, i, e
s/[loves, pete, kath] ⟶ f, q, l
s/[hates, kath, mary] ⟶ x, f
s/[kisses, mary, anna] ⟶ m, m
s/[hates, john, kath] ⟶ z, o
1/anna ⟶ q, d, x
s/[kisses, john, anna] ⟶ j, f, k
s/[hates, pete, anna] ⟶ s, i
s/[loves, mary, anna] ⟶ o, b, e
s/[sees, kath, anna] ⟶ w, l
s/[hates, A, kath] ⟶ 2/A, c
2/anna ⟶ a, d
2/mary ⟶ k, w
s/[sees, john, pete] ⟶ p, p
s/[sees, john, mary] ⟶ o, r
s/[adores, kath, A] ⟶ e, 1/A
s/[hates, kath, pete] ⟶ d
s/[A, pete, john] ⟶ 3/A, d
3/hates ⟶ g
3/loves ⟶ r
s/[A, mary, john] ⟶ d, 4/A
4/sees ⟶ i, z
4/hates ⟶ v

s/[hates, kath, anna] ⟶ i, e, t
s/[hates, pete, mary] ⟶ j
s/[A, anna, kath] ⟶ y, 5/A
5/sees ⟶ i, l
5/loves ⟶ e, t
s/[kisses, pete, kath] ⟶ w, l, g
s/[adores, mary, pete] ⟶ b, y
s/[loves, kath, pete] ⟶ e
s/[kisses, john, kath] ⟶ b, e
s/[kisses, john, pete] ⟶ p
s/[sees, mary, kath] ⟶ y, k, l
s/[sees, A, pete] ⟶ 6/A, i
6/anna ⟶ v, s
6/kath ⟶ r, z
s/[adores, pete, mary] ⟶ z, s, k
s/[hates, anna, A] ⟶ k, 1/A
s/[kisses, mary, pete] ⟶ s, u
s/[loves, john, mary] ⟶ w
6/mary ⟶ c
2/pete ⟶ t, g, c

What is immediately clear is that this grammar is very large, containing a total of 69 rules, almost all of which are holistic and unanalysable. For example, in the rule

s/[sees, mary, anna] ⟶ t, b, s

there is no part of the string *t,b,s* than can be identified as meaning "sees", "mary" or "anna". It is simply the case that the *whole string* has the *entire* meaning [sees,mary,anna].

The grammar is also very suboptimal, it contains a very large number of

rules, 69, and yet is only able to express 63% of the meaning space.

However, already we can see that compositionality is *beginning* to emerge: so far, all the utterances produced have been invented from scratch, as the first agent had no grammar when the simulation began. However, chance similarities between strings had led to generalisations being made, such as those which produced these rules:

s/[adores, A, anna] ⟶ q, 1/A
1/kath ⟶ q, k
1/pete ⟶ j

as would have been formed by the following utterances:

< [adores,kath,anna] q, q, k>
< [adores,pete,anna] q, j>

The fact that the two utterances share 2 out of 3 elements of the semantic vector ("adores" as the *action* and "anna" as the *acted-on*) as well as a portion of the string (the substring *q*) has happened quite by chance, as both strings were generated by random invention. However, it is sufficient to kick-start the generalisation process, resulting in the invention of a new non-terminal *1*. Instances of this non-terminal include the string *q,k* meaning "kath" and *j* meaning "pete".

If we move on approximately ten generations, we can see an increase in the amount of compositionality displayed:

6/loves ⟶ c
s/[adores, A, john] ⟶ z, 3/A, d
6/adores ⟶ z, j
3/john ⟶ j, w, j
s/[hates, A, mary] ⟶ u, b, z, 3/A, g, u, l

6/sees ⟶ d, y, l
4/sees ⟶ w, v
4/loves ⟶ h, s
s/[A, john, B] ⟶ t, 1/A, 7/B
4/adores ⟶ u
s/[A, pete, mary] ⟶ 1/A, z, y, g, u, l
7/mary ⟶ e, d, a
6/kisses ⟶ q
s/[A, kath, B] ⟶ 6/A, 3/B, g
4/hates ⟶ e, g, a
3/pete ⟶ j
3/anna ⟶ y
1/sees ⟶ v, a
3/mary ⟶ e, t
7/anna ⟶ u, b
6/hates ⟶ y, v, u
s/[A, john, mary] ⟶ 9/A, e
7/kath ⟶ r, z
1/adores ⟶ q
s/[A, B, kath] ⟶ 7/B, z, 6/A, l
1/loves ⟶ h, n
s/[A, B, anna] ⟶ 1/A, 3/B
3/kath ⟶ d, n
s/[A, pete, john] ⟶ 6/B, d
9/loves ⟶ o
s/[A, B, C] ⟶ 4/A, 3/C, 3/B
1/hates ⟶ m
7/pete ⟶ m
9/kisses ⟶ m, z
4/kisses ⟶ c, c

In this grammar, which is only a few generations into the simulation, we can see that many more generalisations have been made. In the grammar belonging to the very first agent, the vast majority of the production rules were top level ones, i.e. they had the non-terminal *s* on their left hand side. There were only a few other non-terminal categories. In this grammar, that trend has been reversed: there are many more subrules than top-level ones. Also, in the previous grammar, most of the top level rules were completely holistic, i.e. they contained only terminal characters on their right hand sides, and no variables in their semantic vectors. In the above grammar, however, all of the top level rules have at least one non-terminal on their right hand side, and contain at least one semantic variable relating to this: i.e. components of the meaning have become *lexicalised*. There is even one rule in which *all three* components of the semantic vector have been lexicalised; this rule is maximally compositional and is actually able to express 100% of the meaning space. However, in many cases this particular agent will not use that rule unless there is no other way of expressing the required meaning, because of the deterministic nature of the parser: if there is more than one rule that can be used to express a particular meaning, the first will always be used. Because the rule covering the entire meaning space is quite low down in the grammar, other rules will be selected first, if they exist.

Notably though, despite the much greater degree of compositionality inherent in this grammar, and although it is now able to express 100% of the meaning space, it is still quite large, containing 35 rules. The grammar belonging to the final agent in the simulation is an entirely different story:

s/[A, B, C] ⟶ 2/A, 3/C, 3/B
3/pete ⟶ j
3/mary ⟶ e, t
3/kath ⟶ d, n
3/john ⟶ j, w, j
3/anna ⟶ y

2/kisses ⟶ c, c
2/adores ⟶ u
2/hates ⟶ e, g, a
2/sees ⟶ w, v
2/loves ⟶ h, s

This grammar is fully compositional and entirely optimal: it contains just 11 rules – one for each of the five *individuals* in the meaning space, one for each of the five *actions* and one top level rule specifying how to combine the strings produced by the other rules. This is the minimum number of rules that can be achieved and yet be able to express 100% of the meaning space.

This pattern is a common result in these simulations: a progression is seen from the early agents which have very large grammars able to express only a proportion of the meaning space, to the later ones whose grammars are very small and compact, and able to express 100% of it. This is accompanied by the transition from largely holistic to fully compositional. Figure 4.2 shows the size of grammars versus the proportion of the meaning space they are able to express at the beginning of each simulation, and after 5000 generations. After 5000 generations, all the grammars are able to express 100% of the meaning space, without exception, and 16 of the 25 simulations run have converged on an optimal minimal grammar with only 11 rules, although there are a few simulations which result in grammars of a larger size. It is clear that at the end of the simulation, grammars appear to fall into one of two groups: we will return to this point in section 4.2.1. In general, agents in the simulations become able to express the whole meaning space (or nearly all of it) within a very few generations, as demonstrated in figure 4.3, which shows the proportion of the meaning space expressible by agents in the first 20 generations for 10 typical simulation runs. Most of the simulations have reached 100% expressibility within this timescale. Figure 4.4 shows the size of the grammars for the agents in the first 200 generations of the same simulations. It is clear that it takes a lot longer for grammars to reach

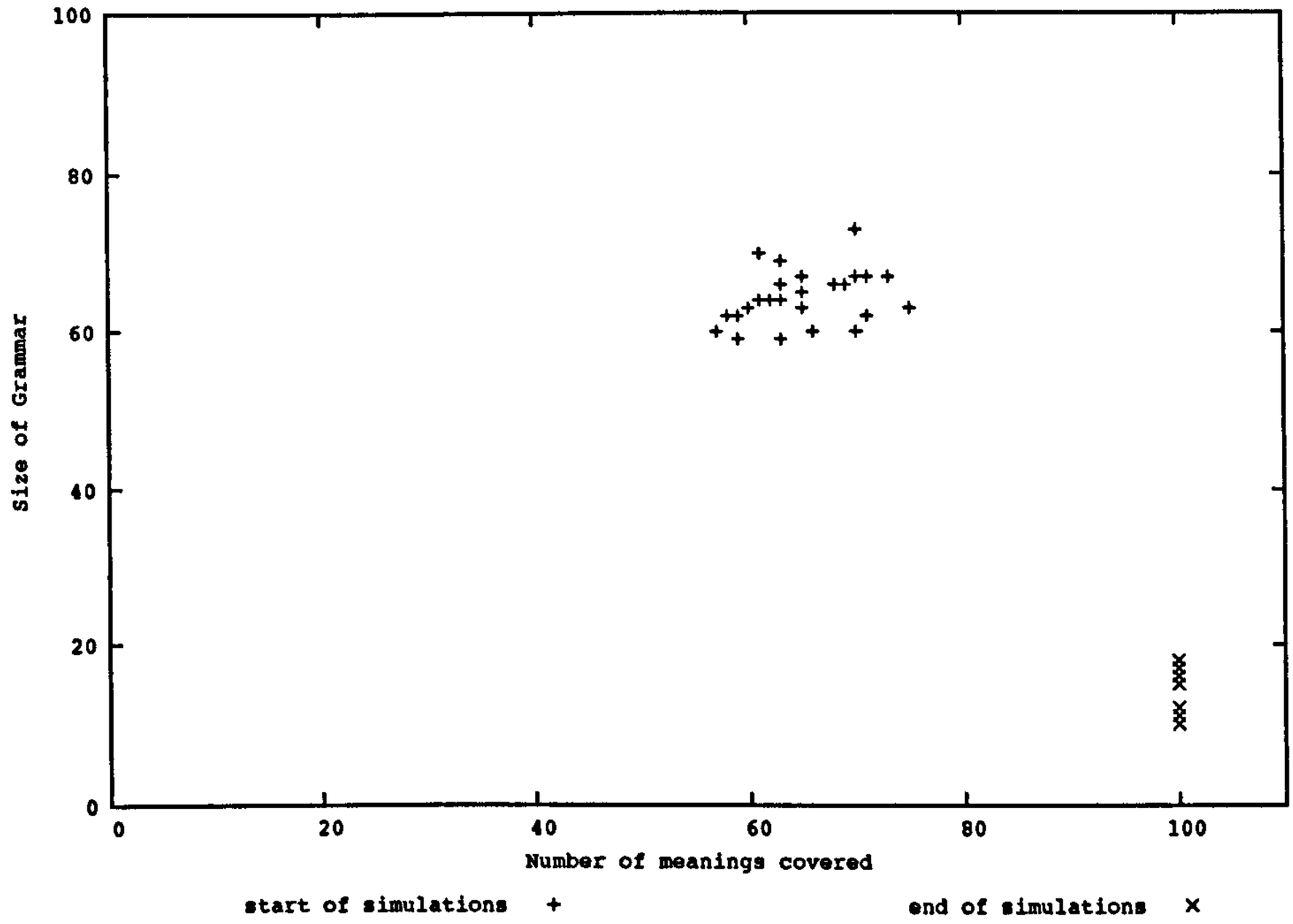

Figure 4.2: The size of grammar versus proportion of meaning space expressible for grammars emerging after 1 and 5000 generations of the simulation.

optimal compositionality (if at all).

As mentioned above, one of the key features of these grammars is the use of word order to specify the distinctions between syntactic categories. For example, in a given sentence it is possible to identify what the subject and object of the event being described are by their positions in the sentence, much as in the English language. Free word order languages do not emerge, and nor does the use of inflection to specify the distinction between thematic roles. This is not entirely surprising, given the nature of the heuristics that are used in grammar induction. These heuristics search for differences between strings: they essentially do so by looking for common prefixes and suffixes. When the parts of the strings which are the same have been identified, those which are different can be attributed to differences in the meanings of the two strings (as explained in Section 4.1.3 which describes the implementation of

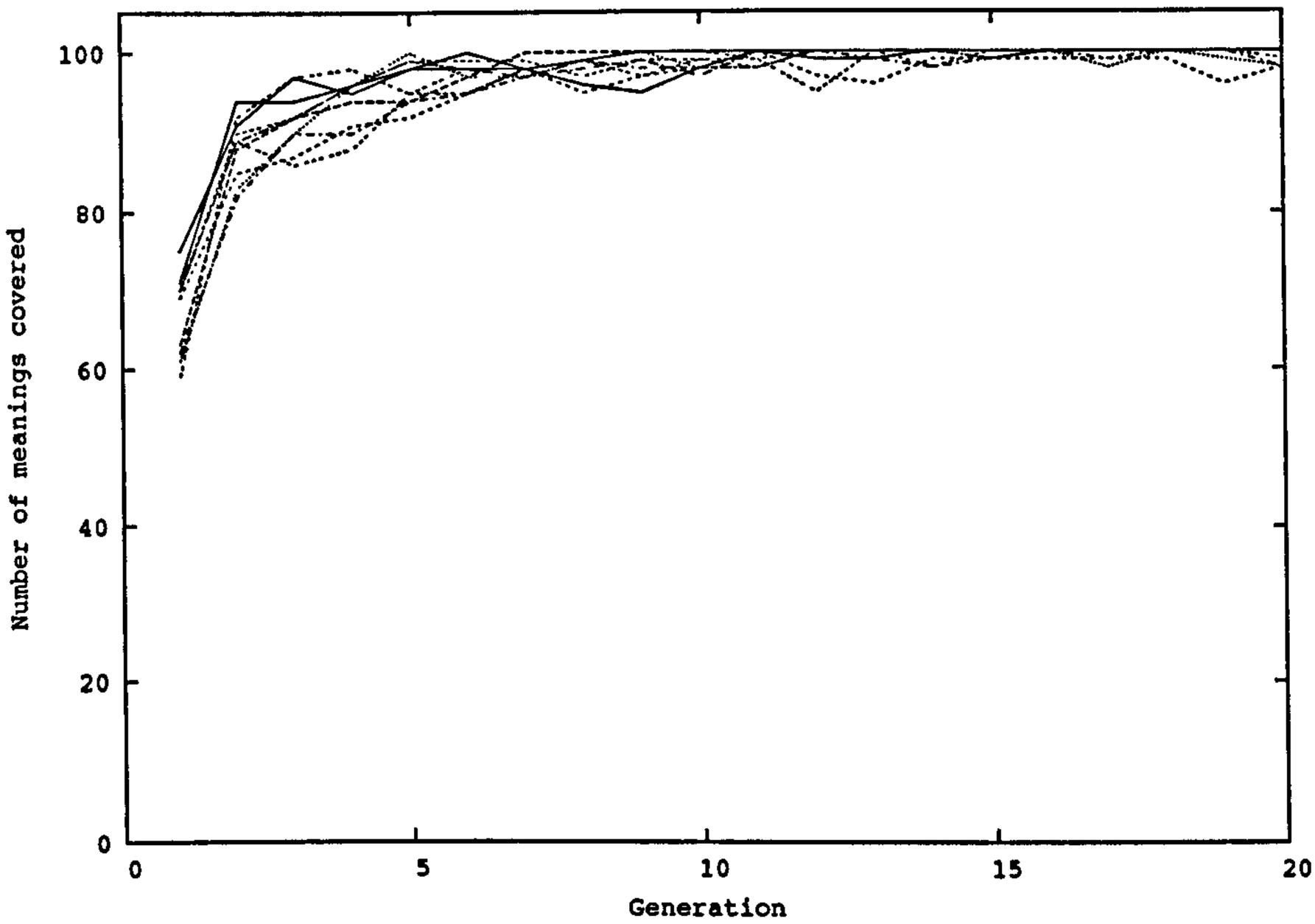

Figure 4.3: The proportion of the meaning space than can be expressed by agents in the first 20 generations.

the grammar induction algorithm). Thus if presented with the strings *abcdef* meaning [loves,john,mary], and *abcdgh* meaning [loves,john,kate], the grammar inducer would identify the common prefix *abcd*, whilst noting that the final sections of the two strings differ. Thus the difference in meaning would be ascribed to this, resulting in the conclusion that *ef* means "mary", whilst *gh* means "kate". Suppose however, that the second string had been *ghabcd*, as might occur in a language that allows freedom of word order. This shares neither a common prefix nor suffix with the string *abcdef*, so the current grammar inducer would fail to notice any similarity between the two. As a result it would not pick out the relevant differences either. Thus if chance regularities between strings did occur such that a free word order language might be induced, the current grammar inducer would not be able to induce it.

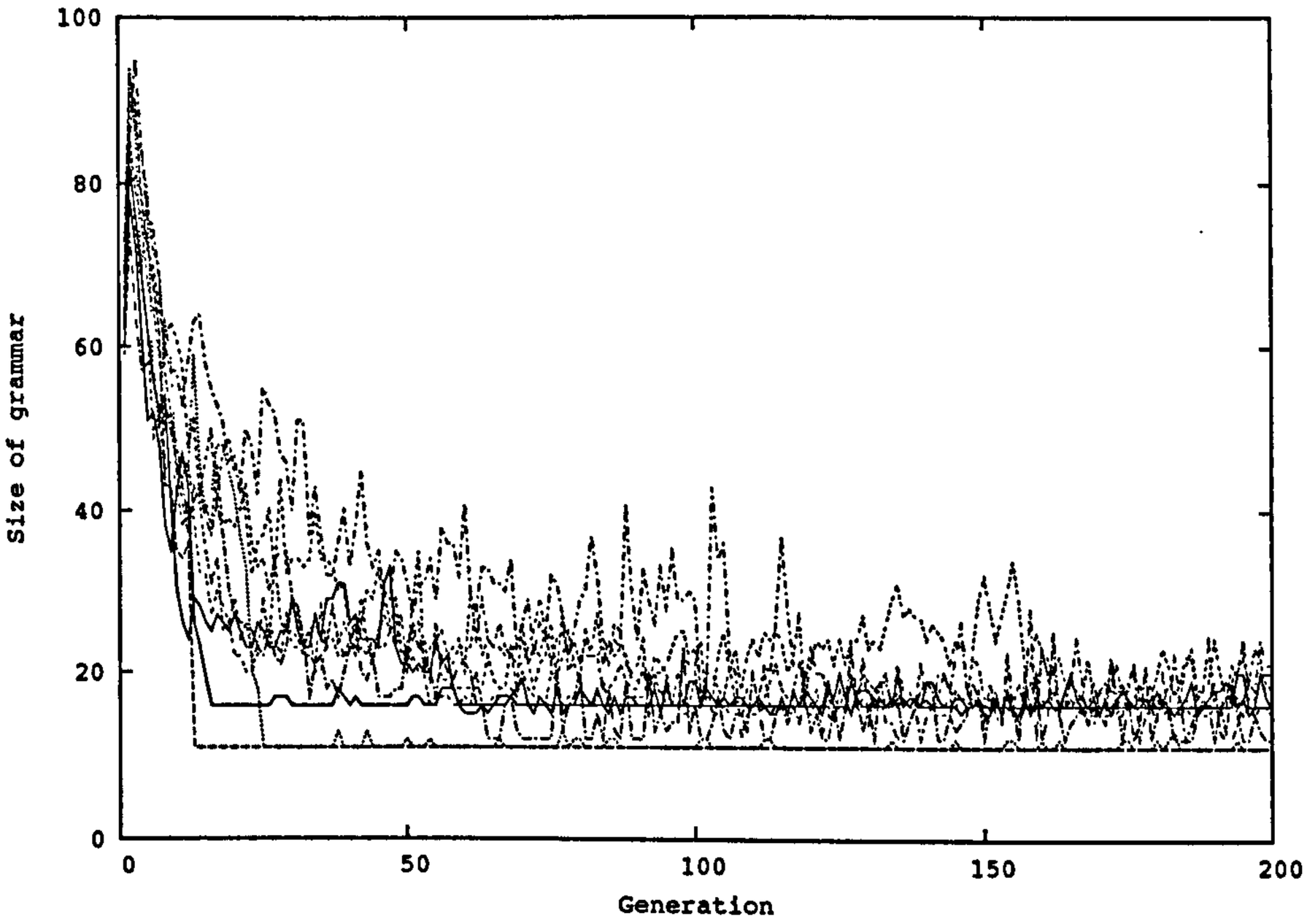

Figure 4.4: The size of grammars belonging to agents in the first 20 generations.

However, natural languages do not tend to exhibit such rigid word order as those emerging from the simulation. Even English, which has a relatively strict ordering, allows a small degree of word order freedom, for example when the speaker wishes to topicalize the object. Other languages, such as German allow a lot more, and still others exist such as Russian which allow almost complete freedom of word order. Clearly, in such languages, it is no longer possible to use word order to distinguish between thematic roles: if the language allows both SVO and OVS sentences, for example, and the string *johnlovesmary* is heard, how is the hearer to distinguish between the two possible meanings [loves,john,mary] and [loves,mary,john]? Instead, inflection is commonly used – different forms of the nouns *john* and *mary* – to determine their case, i.e. whether they are subject or object of the sentence. Thus the two possible meanings can be distinguished by the form of each noun used.

4.2.1 Different subjects and objects

As mentioned briefly above in section 4.2, in some cases the grammars emerging from the simulation did *not* converge on the minimal number of rules, 11, needed to express the entirety of the meaning space. Indeed, a significant number of the runs converged on a larger grammar with 16 rules. The key feature of these grammars is that they contain two distinct noun categories rather than one, one of which is used to express the *subject* of the sentence, and the other the *object*, as in the following example:

s/[P, X, Y] ⟶ 3/X, i, 1/Y, 2/P
1/anna ⟶ i, p, l
1/kath ⟶ c, s
1/mary ⟶ t, a
1/john ⟶ j, e
1/pete ⟶ h
3/kath ⟶ a, k, f
3/pete ⟶ a, u, f
3/mary ⟶ t, s
3/anna ⟶ g
3/john ⟶ p
2/kisses ⟶ t
2/hates ⟶ z, s
2/loves ⟶ m, q, j
2/adores ⟶ u, i
2/sees ⟶ m, y

In this case, the non-terminal category *2* represents the predicate, *3*, the subject and *1*, the object, arranged in the order Subject, Object, Verb. (This grammar also contains a non-terminal category *i* in its top-level sentence rule. This is simply an artifact of the grammar induction algorithm used: as the

different parts of the string get associated with different parts of the meaning, it sometimes happens that some characters get "left behind" and stranded in the top-level rule without being attributed to any particular part of the meaning.)

Could we view such a grammar as exhibiting some form of primitive case system, in that it is possible to distinguish subject forms of nouns from objects, rather than using the same form for both? This is analogous perhaps to highly irregular forms of case found in some languages, such as the English pronouns *I*, *me*, *we* and *us*, where the nominative forms (*I* and *we*) used to the represent the subject of a sentence have no morphological relationship to the accusative forms used for objects (*me* and *us*). Kirby himself makes reference to such a grammar in the intermediate stages of his simulations [45], in which he refers to the two distinct noun categories as "case-marked nominals".

4.3 Summary

In this chapter, the implementation of an iterated learning model has been described, resulting in a successful replication of Kirby's results: the emergence of grammars showing compositional behaviour, in which the meaning of an utterance is a function of the meaning of its parts and the way it is assembled. Like Kirby, we have seen distinct grammatical categories used to express nouns and verbs, and the use of word order to distinguish between different semantic roles. We have also seen the emergence of apparently "sub-optimal" grammars which appear to exhibit two distinct noun categories, one used for the subject of the sentence and one for the object, which might be considered as some form of primitive case system.

The following chapter will describe attempts to create a selective pressure for

languages of this type, in order to see whether it is possible to facilitate their emergence. As previously discussed in Section 3.4, inflectional case endings are commonly associated with languages which display a large amount of optionality in the ordering of words in a sentence: where word order can no longer be relied upon to make semantic roles explicit, alternative cues must be found. The idea here is to introduce a degree of word order freedom into the simulations so that semantic roles can no longer be distinguished on that basis. It is hoped that this will result in pressure for distinguishable subject and object categories, which will in turn promote the emergence of grammars such as the one above.

Chapter 5

The Effect of Word Order

A key feature of the grammars that emerge from Kirby's simulations as described in Chapter 3 and the current implementation of the model described in Chapter 4 is the use of word order to specify meaning distinctions. However, it is worth noting that in many natural languages, meaning distinctions are not wholly specified by word order – even in English, some freedom of word order is allowed, and other languages allow much more. This is generally accompanied by a much richer case system than that found in English.

In Chapter 4, we succeeded in reproducing the results of Kirby's simulations, demonstrating the emergence of compositional languages without natural selection for communicative ability of agents, purely as a result of the adaptation of the language to the dynamics of language transmission and the biases imposed by the learner. Returning to Jackendoff's hypothesis that language may have evolved in an incremental manner [42] discussed in Section 2.4.3, one could perhaps view the outcome of Kirby's simulations and of the current implementation as having reached a level of complexity akin to the "proto-language" stage of development, namely exhibiting properties such as use of symbols in a referential manner, concatenation of symbols, and use of sym-

bol position to convey basic semantic relations. The grammars arising from these simulations could perhaps even be argued to exhibit some of the more advanced features of modern language, in particular the apparent distinction between nouns and verbs, with separate syntactic categories used to express each.

The question is, can this be taken further? Is it possible to encourage the emergence of more of the "later features" of modern language using this framework, and without the need for innate, language-specific knowledge? Kirby has already demonstrated that some degree of behaviour resembling phrase structure is possible in his experiments with recursion [44], and as already mentioned, there does seem to be some evidence for the emergence of syntactic categories. This is similar then to the stage at which Jackendoff suggests that it would have been possible for morphological markings to evolve.

Jackendoff discusses the relationship between the use of word order and the presence of morphological markings, suggesting that they could be viewed as completely separate systems, working in parallel, to accomplish partly overlapping functions. He draws an analogy between this and the perception of depth in the visual system, where there are a variety of disparate mechanisms, all acting in concert to give different kinds of information about the distance from the viewer of the surface being viewed [54]. Under some circumstances, these individual systems provide redundant information, and under others, one or other of them will dominate, and under yet other circumstances still, they may conflict resulting in optical illusions. Furthermore, these systems vary greatly in their evolutionary age, some of them rooted in the more primitive or "earlier" visual areas of the brain, such as the Lateral Geniculate Nucleus, and others dependant on the more recently evolved Visual Cortex.

Jackendoff envisages that the various methods by which a language speaker

might elucidate semantic roles might operate in a similar manner. He considers the use of word order to signal these distinctions to be the oldest method in an evolutionary sense, having developed (as described in Chapter 2) during the proto-language stage. On top of this, though, are built the (sometimes) redundant systems of inflectional marking: verb agreement with subject (and in some languages, object too), and case marking, making up a tri-partite system for signalling semantic roles. As with depth perception in the visual system, the brain is able to make use of these three different systems appropriately according to the circumstances, leaving languages free to "mix and match these strategies in different proportion". The observation above that languages with richer inflectional systems often allow more freedom of word order clearly follows on from this. Once word order is not the exclusive means by which semantic roles may be determined, it can be put to other uses, such as giving information about focus and topic of the discourse. Conversely, in some languages inflection is used for this purpose without any loss of expressivity, because the information about semantic roles can still be conveyed by other elements of the tri-partite system.

In this chapter and those that follow, we will be attempting to build on the use of word order to make semantic distinctions that can be observed in the results of Kirby's studies and in the current replication of those experiments described in Chapter 4. The aim of this work is to see if it is possible to add a second part of this tri-partite system within the current framework: inflectional case markings. The hypothesis being explored here is whether or not such a system can emerge with the cognitive machinery these agents already have, developing as a result of their "general purpose learning abilities", helped along by the "dynamics of language transmission", or whether, in fact they need some kind of LAD in the form of innate language-specific knowledge in order for this behaviour to be possible.

5.1 The Need for a Non-Deterministic Parser

It is observed in the replication of Kirby's system described in Chapter 4, that "suboptimal" grammars occasionally emerge which contain two separate syntactic categories denoting the *individuals* in the meaning space, one of which will be used for the subject of the sentence and the other for the object. These separate subject and object noun categories can perhaps be viewed as a primitive form of case system. The work described in this chapter is an attempt to promote the emergence of grammars with such properties.

This endeavour will involve the introduction of a degree of word order flexibility to the Iterated Learning Model described in Chapter 4. This will take the form of the occasional re-ordering of the elements of a sentence, as might occur in natural language when a speaker topicalises a word for emphasis, or even makes grammatical error. When such re-ordering occurs, it is hoped that the sentence exhibiting the alternative word order will be learnt by the agent to which it was spoken, and incorporated into that agent's grammar as an alternative means by which the meaning in question may be expressed. It is anticipated that the acquistion of alternative word orders in this way might create a selective pressure for the emergence of distinguishable subject and object nouns, resulting from a need to disambiguate potentially conflicting word orders. For example, if a language permits both SVO and OVS orderings, then it will be impossible to determine the subject and object of any given sentence unless different noun forms are used for each.

However, the current model does not really support the use of multiple word orders, due to the deterministic nature of the parsing and production algorithm employed: if a learning agent *observes* an alternative word order which is not already encompassed by its grammar, it will find that it is unable to parse the utterance in which it was observed, and will use that utterance as the basis for induction. Thus the appropriate rules will be added to its gram-

mar. As a result, in future it will be able to successfully parse any utterance it encounters which exhibits this new alternative order. However, during production, because the agent will simply scan the database for the first set of rules capable of producing a string with the required meaning, any subsequent sets of rules will never be used, and thus sentences with other word orders will never be generated. Therefore, although alternative word orders can be *learnt*, enabling utterances that are examples of these word orders to be parsed, they will not be passed on from one agent to the next during the process of cultural transmission. Thus it is necessary here to make changes to the parser in order for an agent to be able to have more than one word order in its grammar, and to be able to *use them all* productively.

5.1.1 Random selection of strings

Perhaps the simplest way to achieve this would be for the agent to select *all* possible ways of expressing a given meaning, and to choose one of them at random. However, attempts to implement this tend to result in simulations which do not converge on the minimally sized but maximally expressive grammars seen in Chapter 4, but instead result in grammars containing a very large number of rules, and a very large number of alternative ways of expressing a given meaning. In the system using the strictly deterministic parser, grammars have generally achieved a fairly small size within the first 100 generations, the average size at this stage being approximately 19 rules. However, using this simple method to enable the expression of alternative word orders raises the average number of rules after 100 generations to 61.6. In both cases most grammars are already able to cover 100% of the meaning space, although some of those using the new production algorithm do fail on that score. Figure 5.1 shows grammar size versus the number of meanings covered after 100 generations for both the new and the old production algorithms. It can clearly be seen that simulations employing the original

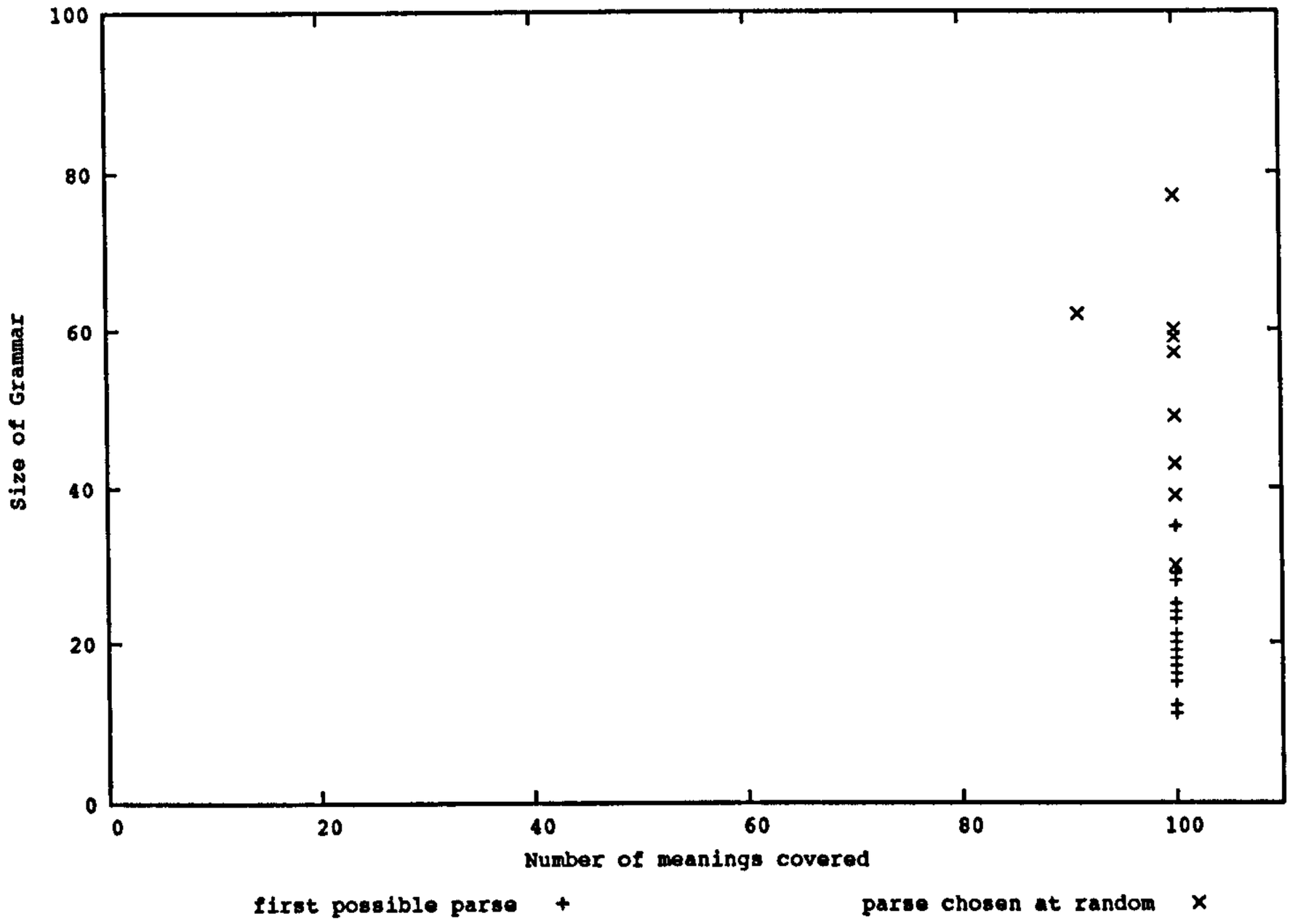

Figure 5.1: *A scatterplot showing the sizes of grammars vs the number of meanings covered at 100 generations for the version of the model that uses the first available string for each meaning and that which selects a string at random from amongst all possible utterances. In both cases is it possible to see that most grammars cover 100% of the meaning space, but the size of the grammars differs wildly in the two cases.*

deterministic production algorithm, in which agents always use the first available string for a given meaning as described in Section 4.1.2, result in much smaller grammars than those using the new version where agents generate all possible strings and select one of them at random.

The grammars that are being transmitted from one generation to the next when the new production algorithm is used start large and stay large, many of them becoming very unwieldy, and resulting in simulation runs that grind to a halt and fail to reach the requisite number of generations (which in these experiments is taken as 5000). There are simply *too many* possible strings

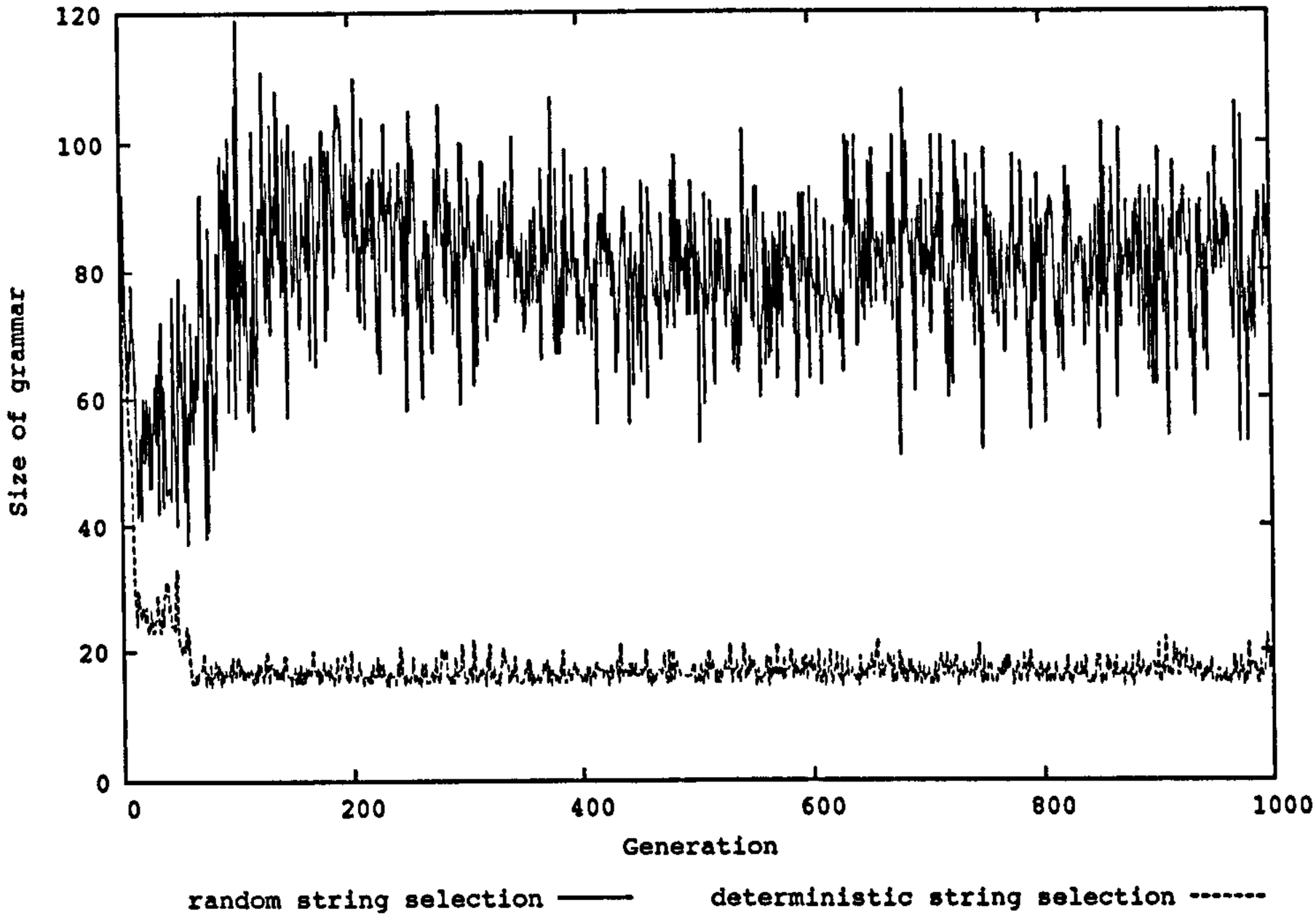

Figure 5.2: A graph showing the size of the grammar at each generation for one run of the simulation when using the version of the model that chooses one of the possible strings for a given meaning at random. A comparable plot for the purely deterministic production algorithm is also shown.

for an agent to be able to enumerate them all. Even those simulations which do complete fail to converge on a minimal grammar. Figure 5.2 illustrates this by showing the size of agents' grammars at the end of each generation, for the first 1000 generations of one such run. There is a short initial period during which the number of grammar rules starts to decrease, (this only occurs within the first few generations and is almost unnoticeable on this graph), but it is swiftly followed by a dramatic turnaround. From about generation 20 onwards, the number of grammar rules increases rapidly again. This behaviour is typical. The plot also shows the results from a run of the deterministic parser for comparison.

5.1.2 Random selection of rules

To address the issue that many of the simulations were failing to complete as a result of there being far too many possible strings for an agent to enumerate, an alternative approach to allowing the propagation of alternate word orders was attempted. In the original parsing and production algorithm, the *first* rule in the grammar with the correct non-terminal on its left hand side and semantics that can be unified with the desired meaning is chosen for expansion. In the revised version described in Section 5.1.1, *all* such rules in the grammar are selected and expanded, and one of the strings returned is chosen at random. In the alternative to this, again all the rules are selected, but rather than expand them all, just one is chosen at random. Thus, for example, if the desired meaning were [sees,kath,anna], then all rules which have the non-terminal *s* on their left hand sides, and whose meanings can be unified with [sees,kath,anna] will be picked up. This might include completely holistic rules, such as

s/[sees,kath,anna] $\longrightarrow$ o, u, z

or rules which have been completely lexicalised, such as

s/[A,B,C] $\longrightarrow$ 2/A, 3/C, 3/B

(as the variables *A*, *B* and *C* are unifiable with the atoms *sees*, *kath* and *anna*) or rules which have been only partially lexicalised, such as

s/[A, anna, kath] $\longrightarrow$ y, 5/A

Once these rules have been identified, one of them is chosen at random, and the process is repeated for any non-terminals on the right hand side of the selected rule. Thus in our example, if the rule chosen were the second one,

s/[A,B,C] $\longrightarrow$ 2/A, 3/C, 3/B

the variables A, B and C will become unified with the meaning [sees,anna,kath], and the agent will go on to seek out rules with the non-terminal *2* on their left hand side and whose meanings can be unified with the atom *sees*, etc. If at any stage a chosen rule contains non-terminal categories on its right hand side that cannot be expanded, it is removed from the list of potentially applicable rules, and a new one is selected, again at random. Rules are picked with equal probability.

Unfortunately, this type of non-deterministic implementation also results in simulation runs which fail to converge on the minimally-sized but maximally expressive grammars seen when the original deterministic production algorithm is used. Figure 5.3 shows the size of grammars versus the number of meanings that can be expressed after 100 generations, for both the original model in which agents always use the first available string for a given meaning (i.e. the deterministic system) and the new version in which one rule is chosen at random for expansion from amongst all possible rules at each level of the parse tree.

Again, grammars tend to be very large and to contain very many ways of expressing a given utterance. Their size seems to be similar to that found when selecting strings at random, with the average number of rules after 100 generations being approximately 56. However, the *lengths* of the rules seems to be even greater. One particular feature of both types of grammar is the presence of *very long* rules, each containing large numbers of terminal characters, often including repeated segments. Many of these rules differ from each other only by the number of repetitions of these segments. Similar problems were also encountered by Smith and Hurford [71] in their attempts to extend the Iterated Learning Model to populations of multiple agents.

In both cases, failure to converge when using a non-deterministic parser is

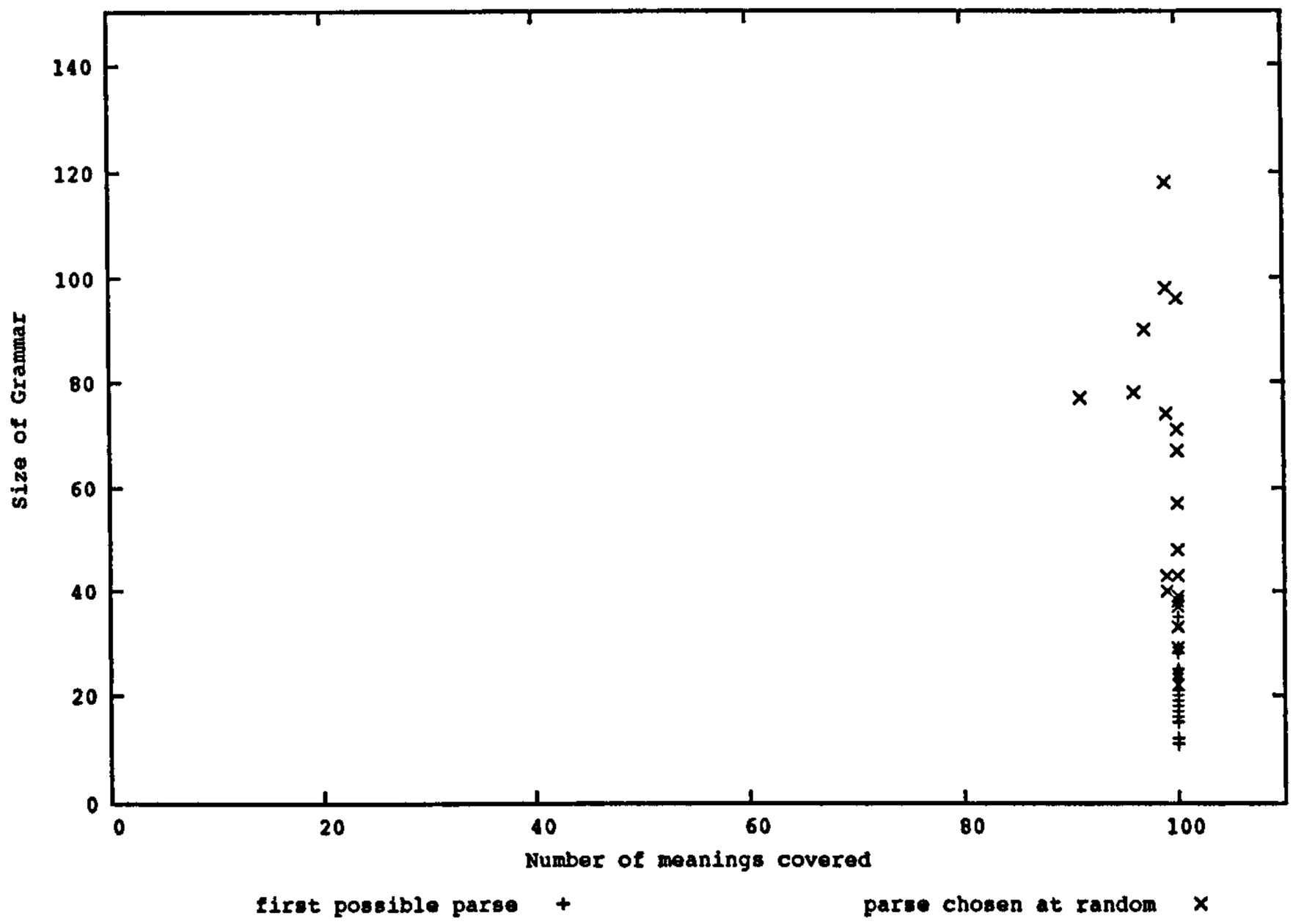

Figure 5.3: A scatterplot showing the sizes of grammars vs the number of meanings that can be expressed at 100 generations for the version of the model that uses the first available parse for each meaning and for that which chooses one of the possible parses at random.

presumably related to Smith's result [70] that in order for compositionality to emerge in an Iterated Learning Model, a one-to-one correspondence between meanings and strings must be established. In the original (deterministic) implementation of the system described here, this property is established by two means:

- a learner will not seek to associate more than one meaning with a particular string (i.e. if it can parse a string, it does not use it as input for the induction process, even if the meaning returned by the parse was not that intended by the speaker).

- if there is more than one way of expressing a particular meaning, agents

will always choose the same one, due to the deterministic nature of the parsing and production algorithm, which simply scans the rules in the database until it finds the first set that can produce a string for the required meaning.

Clearly this is not a property of *natural* languages, where synonymy and homonymy are common occurrences. However, Smith argues that children *do* have such a bias, and quotes Slobin [68] in saying that they prefer a one-to-one mapping of content and form wherever possible.

This requirement leads to a fundamental quandary. In order to introduce freedom of word order as a feature that might promote the emergence of distinguishable subject and object categories, we need agents that are capable of understanding, and productively using multiple versions of a particular meaning. This is the only way in which alternative word orders, once introduced, can be passed on to subsequent generations. And yet, in order for a grammar to emerge at all, we need agents that maintain a one to one correspondence between meanings and strings.

5.1.3 A quasi-probabilistic parser

How strong is the requirement for a one-to-one mapping between meanings and strings? Must the same meaning *always* be expressed by the same string, or is it sufficient if it *usually* is? With this in mind, a form of probabilistic parsing and production algorithm was developed, based on the second non-deterministic one above: at each level of the sentence generation process, every rule that might be capable of producing the required meaning is once again selected, and one of these rules is chosen for expansion, as in Section 5.1.2. However, the rules are no longer chosen with equal probability, but in proportion to the number of times each has been used previously.

This was implemented by associating with each rule a count, representing the number of times it has been used; the probability that a given rule will be chosen is proportional to the size of that count. Rules are selected as follows:

- Find *all* rules in the database with the appropriate non-terminal category on their left hand sides whose semantics can be unified with the semantics of the intended utterance.
- Retrieve the count associated with each of these rules.
- Assign a numerical range to each rule, which has the same magnitude as the count associated with that rule. Ranges are consecutive; thus if the first rule spans the integers from 0 to 5, the range for the second will start at 6.
- Select at random an integer between 0 and the highest value associated with the last rule.
- The chosen rule is the one whose range that integer falls into.

Thus if a rule has a very low count, it will have a very small range associated with it. This will result in a low probability that the integer chosen will fall into that range, consequently giving a low probability of the the rule being chosen for use.

As with the production algorithm described in Section 5.1.2, if a rule chosen at any stage in the process cannot be expanded any further, it is removed from the list of potentially applicable rules, so that an alternative may be selected. This requires the ranges associated with the remaining rules to be adjusted to compensate for the missing rule. A new integer is then chosen at random.

This method of rule choice has an advantage over the deterministic production algorithm in that it gives agents the ability to use multiple strings to express the same meaning, and thus will allow grammar rules for sentences with different word orders to co-exist, *and* to be used in production, enabling agents to pass on those word orders to subsequent generations. It has an advantage over the the previous non-deterministic implementation of the model where rules are chosen at random from the database with equal probability, because it increases the chances of the same rule being used to generate a string for any given meaning much of the time. If there is a dominant rule with a high count, then this is the one that will usually be used, whilst a rule for a sentence structure with an alternative word order will be used less frequently. Thus, whilst not committing agents to a one to one mapping between strings and meanings, which obviously is not possible if the grammar is to tolerate alternative word orders, it does preserve this in a slightly weakened form by ensuring that the same string is *usually* used to express a given meaning.

But does it allow the emergence of compositional grammars, whilst also making it possible for agents to understand and productively use rules for multiple word orders?

Inserting the new parsing and production algorithm into the original simulation without introducing any freedom of word order at this stage appears to show that the answer to this question is "yes". The majority of simulations (96.15%) using the new quasi-probabilistic parsing/production algorithm converge on minimal sized maximally expressive grammars of the type seen in the output of Kirby's simulations [43]. There is still a slight issue with the insertion of repeated sequences into rules (as in the two non-deterministic attempts described in Sections 5.1.1 and 5.1.2). Again, this results in grammars with large numbers of very long rules, as described above, but these are comparatively rare (occurring in approximately 1 in 25 runs). It would appear that this type of behaviour is associated with the breakdown of one-

to-one mapping between strings and meanings. What is apparent about the grammars that behave in such a manner is that they contain very many rules for each syntactic category, each tending to have very low counts associated with them. This results in very many possible ways of producing an utterance for a given meaning, all with a fairly similar probability of being chosen.

A snippet of such a grammar is given below, taken 100 generations into a simulation:

s/[P,X,Y]	⟶	5/P,l,u,4/X,u,4/Y	2
s/[P,X,Y]	⟶	5/P,l,u,4/Y,u,4/X	2
s/[P,X,Y]	⟶	5/P,l,u,4/X,u,p,y,f,u,p,y,f,u,4/Y,u,p,y,f	3
s/[P,X,Y]	⟶	5/P,l,u,p,y,f,u,p,y,f,u,p,y,f,u,4/X,u,4/Y	7
s/[P,X,Y]	⟶	5/P,l,u,p,y,f,u,p,y,f,u,4/Y,u,4/X,u,p,y,f	2
...			

Here we can see a number of rules with fairly low count associated with them (the highest in this case is 7), which are all essentially very similar. Each takes a word of non-terminal category *4* as its subject, another of non-terminal category *4* as its object, and a word of non-terminal category *5* as its predicate. The rules differ primarily in the number of repetitions of the sequence *u,p,y,f* inserted at various points in the string. The also differ in word order, sometimes showing a VSO pattern, and sometimes a VOS.

Looking at the entries in this grammar for words on non-terminal category 4, we can see that the situation is similar:

4/john	⟶	y,f,u,p,y	28
4/john	⟶	p,y,f,u,y,u,p,y,f	14
4/john	⟶	p,y,f,u,p,y,f,u,y,u,p,y,f	2
4/john	⟶	p,y,f,u,p,y,f,u,p,y,f,u,p,y,f,u,y	20
4/pete	⟶	p,y,f,u,p,y,f	23
4/pete	⟶	p,y,f,u,p,y,f,u,p,y,f,u,p,y,f	52
4/pete	⟶	p,y,f,u,p,y,f,u,p,y,f,u,p,y,f,u,p,y,f	2
4/pete	⟶	p,y,f,u,p,p,y,f,u,p,y,f,f,u,p,y,f,u,y,u,p,y,f,u,p,y,f	3
4/mary	⟶	w	76
4/kath	⟶	t	56
...			

Here we can see that of the five *individuals* in the meaning space, two of them, *john* and *pete*, have multiple entries within this syntactic category. It is possible to use any of these entries in conjunction with the rules above, and although the counts associated with them are variable, there is no clear *dominant* rule which is likely to be chosen in the majority of cases. Thus there are very many possible ways of expressing any meanings from the meaning space involving those two individuals, each with a fairly similar probability of being used. And this is only a part of the grammar for this agent – the full grammar contained a total of 85 rules, of which 46 were top level rules for composing sentences. The highest count on any of the top level rules was 20. Later generations in the same simulation contained even larger numbers of even longer rules with even lower counts.

However, as previously mentioned, the incidences of grammars such as these are comparatively rare, and on the whole simulations do converge on small, expressive, compositional grammars, with an overall pattern of results similar to that achieved using the deterministic parser. Figure 5.4 is a scatterplot showing the size of grammars versus the proportion of the meaning space they are able to express at the beginning of the simulation runs, and after 5000 generations, using the new probabilistic parsing and production algorithm.

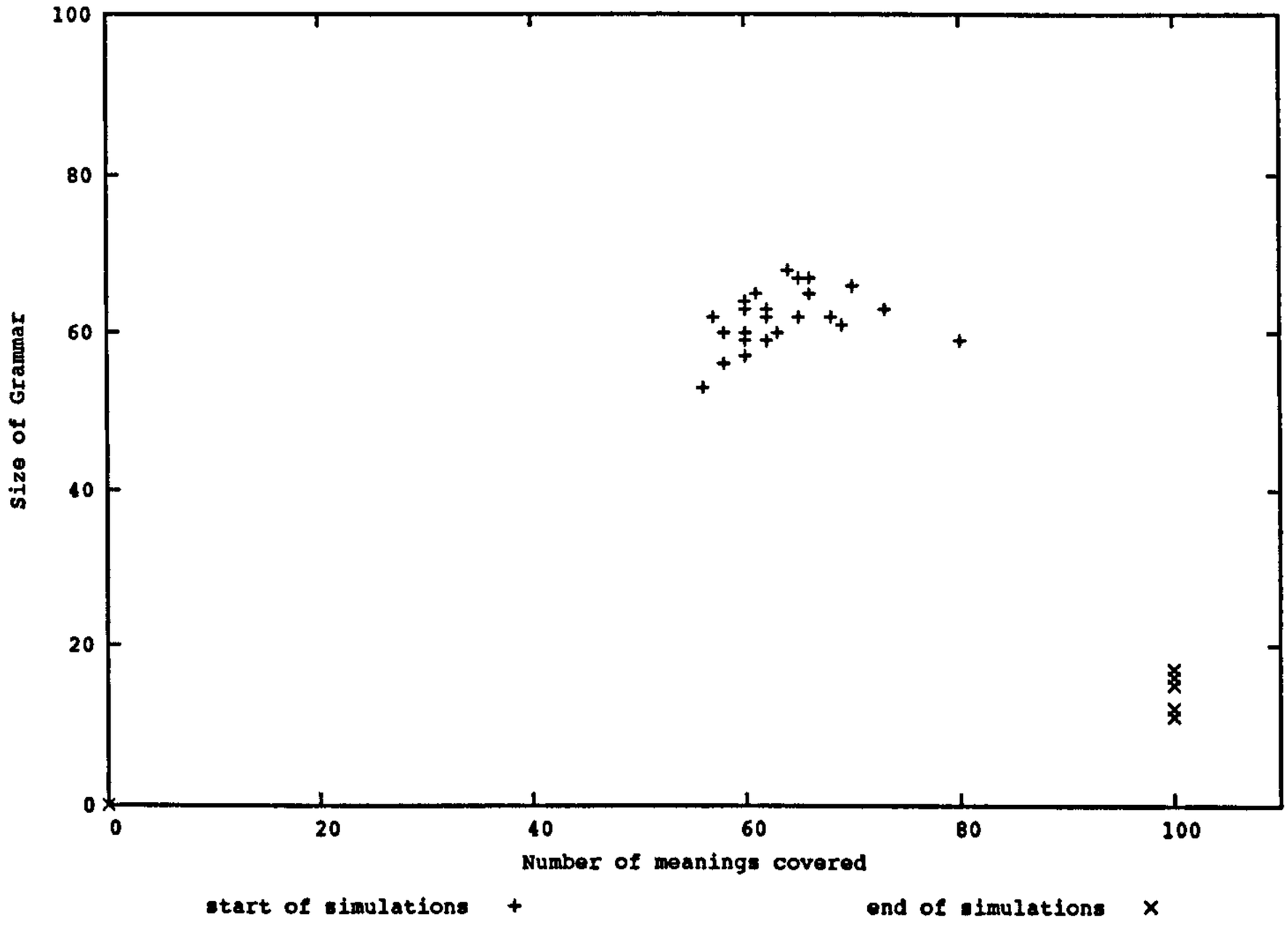

Figure 5.4: A scatter plot showing size versus proportion of meaning space expressible for grammars emerging after 1 and 5000 generations of the simulation using the probabilistic parser.

This is clearly a very different situation to our earlier attempts to allow multiple word orders to exist and be productively used: once again, we can see that the grammars of agents from early in the simulations generally have quite a large number of rules, and yet are able to express only a proportion of the meaning space, whilst the grammars belonging to the final agents can all express 100% of the meaning space, with far fewer rules, indicative of compositional behaviour. Many of the grammars contain only the minimal number of just 11 rules, and even though some of them are larger, they all contain fewer than 20.

Figures 5.5 and 5.6 show that although the number of generations taken to achieve a grammar capable of expressing the entire meaning space is similar when the quasi-probabilistic parser is used instead of the deterministic one,

the length of time taken to achieve a grammar with the minimal number of rules tends to be longer, as can be seen when comparing figure 5.6 with figure 4.4. This is unsurprising; early on in the simulations there are often multiple ways of expressing different meanings, but these tend to die out fairly quickly when using the deterministic parser. The ordering of the rules in the database is the crucial thing here: whichever rule is first will *always* be the one chosen, and alternatives will not be propagated unless they are essential for the expression of another meaning. When using the probabilistic parser, these rules are much more likely to be propagated for longer. However, those rules which are less compositional and therefore can be used to produce strings for a smaller range of meanings will be less likely to be chosen. This will result in their having a smaller count relative to more "useful" rules in the next generation, and being even less likely to be chosen when that agent becomes speaker. Thus the pruning of alternative ways of expressing a given meaning still occurs, but much more slowly. It is not the one-step process seen when using the deterministic parser.

Further examination of the grammars output from simulations using the new probabilistic parser, shows that the most common outcome is a single syntactic category used to express verb-like concepts, and another to express noun-like concepts: this occurs in 48.0% of runs. The other pattern, consisting of a single verb-like syntactic category, plus *two* noun-like categories, one of which is used to express the subject of the sentence, and the other the object, also occurs in 33.0% of cases. The remaining 19.0% of runs fail to converge on a grammar of either type.[1] Comparing this to the output of simulations using the deterministic parser, we find that 58.33% of these result in grammars containing a single verb-like category, and a single noun-like category, 19.05% contain *two* noun-like categories, and 22.62% fail to

[1]A grammar is deemed to have converged by looking at all fully lexicalised top level rules (i.e. all those with only variables in their semantics) and determining whether they all contain the same terminals and non-terminals. Rules which are not fully lexcalised are discounted because even in otherwise optimal grammars, their occurrence is very common.

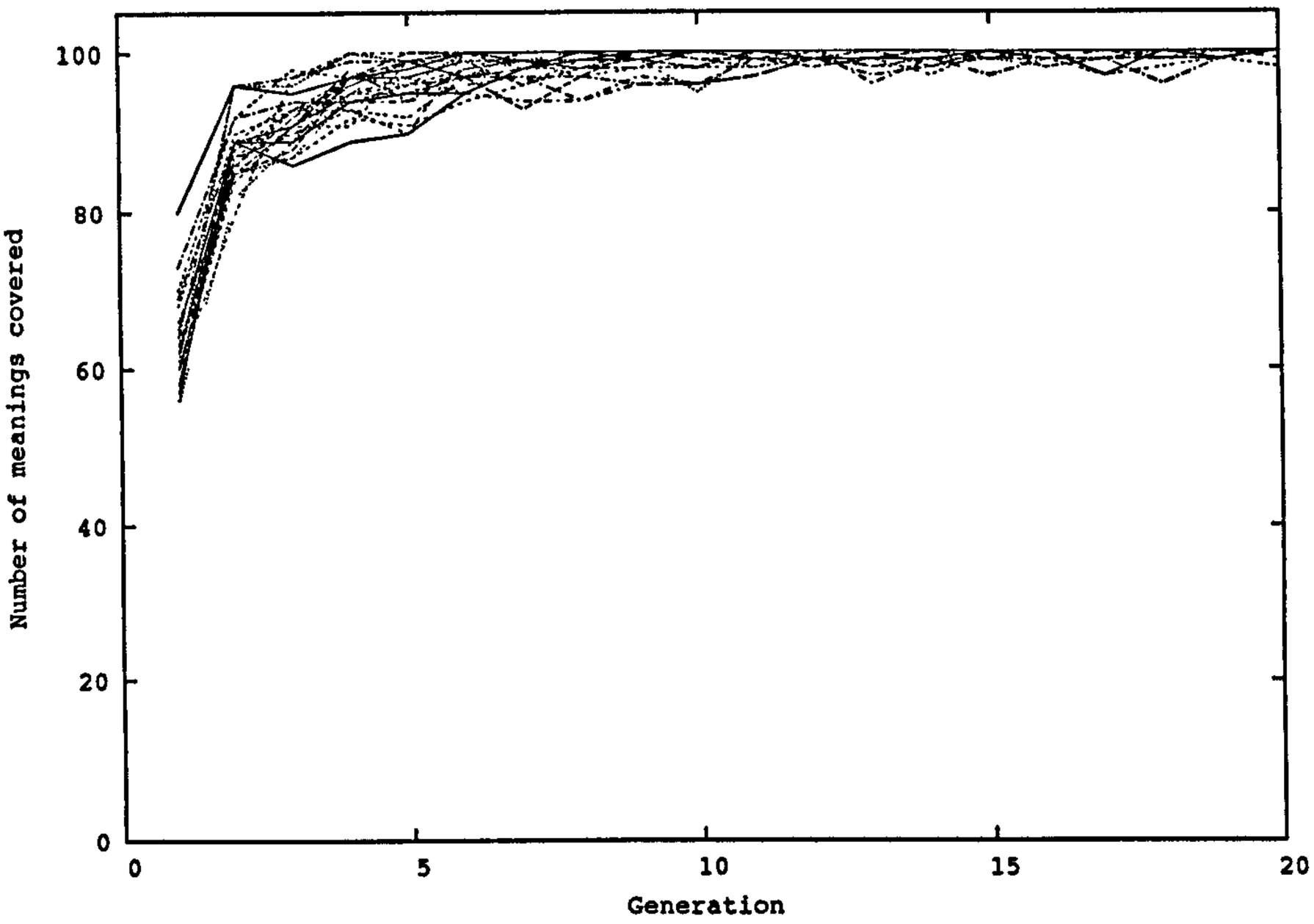

Figure 5.5: *The proportion of the meaning space than can be expressed by agents in the first 20 generations for simulations using the probabilistic parser.*

converge on either.

Thus it is interesting to note that simply enabling the expression of multiple word orders by making the grammar probabilistic rather than deterministic is sufficient to encourage the emergence of the feature which we would like to promote: that is, distinguishable subject and object forms for each individual in the meaning space, the incidence of which has increased from 19.05% to 33% of simulations. Comparing the proportions of 1 noun and 2 noun grammars in runs using the original deterministic parser and those using the new probabilistic one shows this difference to be a statistically significant difference (i.e. $p<0.05$).

Why does this happen? Perhaps alternative word orders are emerging spontaneously, and thus creating a selective pressure for two distinct types of

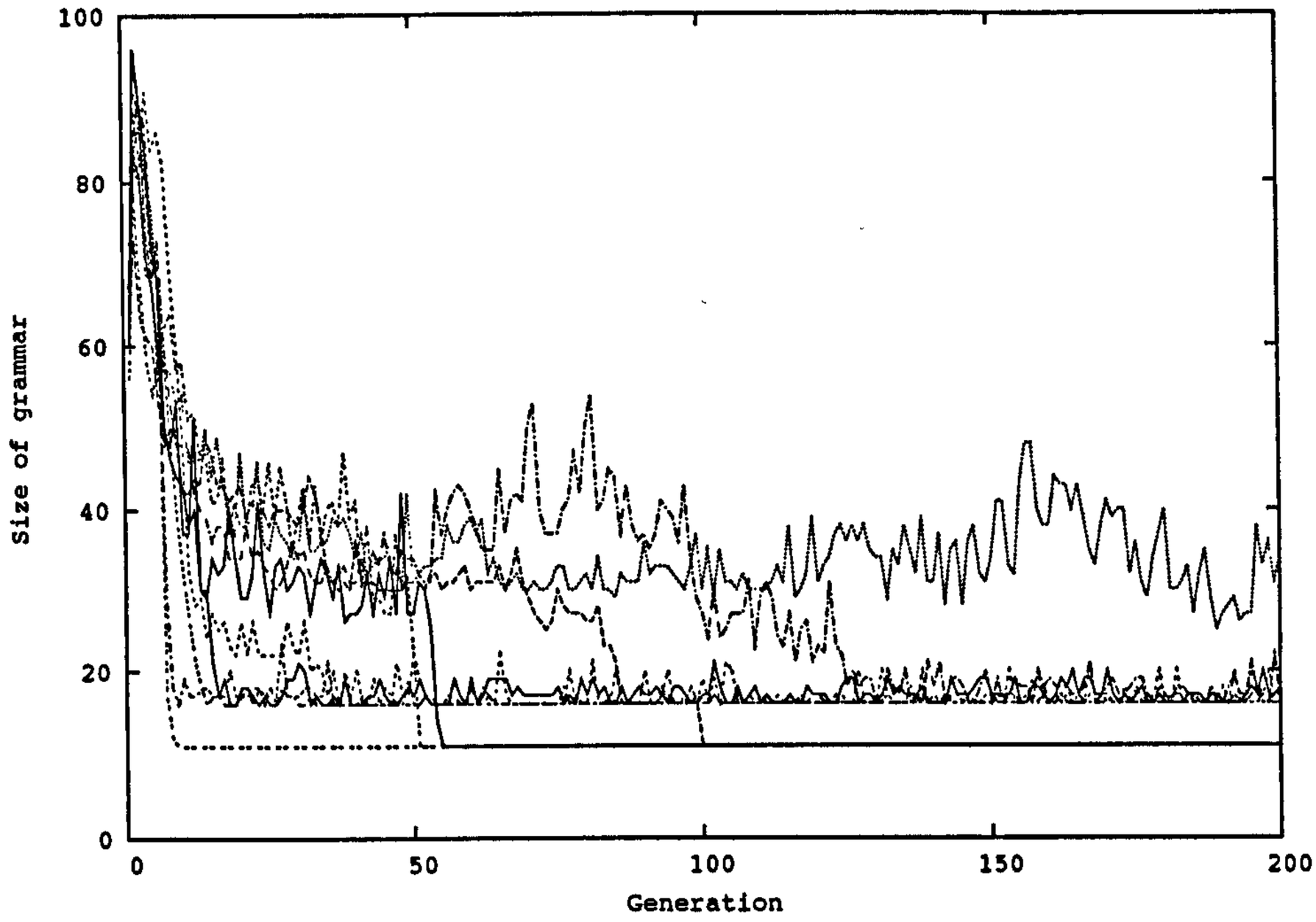

Figure 5.6: *The size of grammars belonging to agents in the first 20 generations for simulations using the probabilistic parser.*

noun category? Those grammars emerging from the system which contained two noun categories were examined to see if this was the case. Of the 33 such grammars, 9 showed spontaneous reordering (approximately 27.27%). This is very similar to the situation when the deterministic parser is used: although fewer grammars have two noun categories, a similar proportion of those that do, i.e. 4 out of 12 (33.33%) exhibit spontaneous reordering. Thus this does not appear to be a factor in the emergence of this grammar type. What is noteworthy is that in every case of spontaneous reordering, whether produced by the deterministic or the quasi-probabilistic parsing and production algorithm, the thematic role expressed by the two noun categories appears inconsistent between the different rules. For example, if there are two top level rules specifying how to construct a sentence, a given noun category will be used in one of them as the subject of that sentence, and in the other as the object. Furthermore, in each of these cases, it appears that at least one

of the noun categories lacks a lexical entry for at least one *individual* in the meaning space. Thus the second rule specifying alternative word ordering is able to *compensate* for this deficiency in the grammar, by allowing *the other* noun category to assume the semantic role (either subject or object) of the missing lexical entry. For example, a grammar may contain noun categories NT1 and NT2, and a rule which relates them in such a way that NT1 is used for the subject and NT2 is used for the object. However, it may be the case that category NT1 lacks an entry in that grammar for the individual "john". In this situation, another rule will commonly be found in which NT1 is used as the *object* and NT2 as the *subject*, which thus allows meanings in which "john" is the subject to be expressed. An example of such a grammar is given below:

s/[P,X,Y] ⟶ 1/X, 2/P, 3/Y
s/[P,X,Y] ⟶ 1/Y, 3/X, 2/P
1/anna ⟶ p
1/mary ⟶ i
1/kath ⟶ g, q
1/pete ⟶ b
2/kisses ⟶ a
2/sees ⟶ p, y
2/hates ⟶ s
2/adores ⟶ c, p
2/loves ⟶ c, i, n
3/kath ⟶ n, w
3/pete ⟶ l
3/john ⟶ t
3/mary ⟶ g, y, x

In this particular grammar, *both* the noun categories are unable to express the full range of individuals in the meaning space. Category *1* lacks a lexical

entry for the individual "john" and category *3* lacks "anna". The first top level rule in the grammar indicates that category *1* should be used as the subject of the sentence, category *3* as the object, and that they should be combined with a verb of category *2* in a SVO ordering. Due to the lack of lexical entries for "john" and "anna", it is not possible to use this rule to express any meaning in which "john" is the subject or "anna" the object. However, the presence of a second rule overcomes this, by allowing Category *1* (which does include a lexical entry for "anna") to act as the *object* of the sentence and using Category *3* (which does have a lexical entry for "john") as the *subject*, combined in a OSV ordering. The appearance of this alternative word ordering in order to overcome the missing lexical items actually occurs very early in the simulation from which this grammar was taken: at generation 123. Interestingly at this stage, a lexical entry of Category *1* for the individual "john" does actually exist, although Category *3* is lacking entries both for "anna" and "kath". Not only does the alternative word order compensate for missing lexical entries, but it means that it is no longer crucial to have a full complement of lexical entries for both noun categories, making it possible for some of them to be discarded in this way without loss of expressivity.

Thus it would appear that this spontaneous reordering emerges as *result* of the existence of two separate noun-categories, rather than promoting it, by making it possible for a speaker to compensate for items missing from its lexicon.

5.2 Introducing Word Order Freedom

Now that we have created a system where agents are capable of understanding and productively using multiple word orders for a given sentence, what happens when we introduce a degree of word order freedom? To do this, a

very simple "re-order" function was added to the system which would reorder the parts of the sentence produced by the speaker before conveying it to the learner. This is done with a fixed probability p. In the early stages of the simulation, where the grammar is composed entirely of idiosyncratic strings, this will have little effect other than to scramble those strings, whereas later, once separate syntactic categories have begun to evolve, it will have the result of re-ordering the subject, object and verb components of the sentence. This is intended to model the occasional "mistake" or use of word order to provide emphasis by the speaker. Once an utterance with an alternative word order has been made, if it is incorporated into the learner's grammar, the rules producing that word order may well be used again, when that agent becomes a speaker. Thus it may be propagated down the generations. It is hoped, that by generating potentially ambiguous word orders, this will act as a driving force towards distinguishable subject and object versions of each noun, i.e. a primitive case system.

The rationale behind this is that occasionally the re-ordering operation may cause the position of the subject and object nouns to be inverted, resulting in the meaning of the sentence appearing to be something quite different to that intended. So, if an agent already has a the string *j,o,h,n,l,o,v,e,s,m,a,r,y* in its grammar, associated with the meaning [loves,john,mary], it may go on to hear the same string meaning [loves,mary,john] (having been produced by the inversion of subject and object by the re-ordering operation on the original string *m,a,r,y,l,o,v,e,s,j,o,h,n*). Because agents are biased towards associating only a single meaning with a given string, under such circumstances, the *new* meaning of the string *johnlovesmary* would never be learnt. However, had the original utterance (prior to re-ordering) meaning [loves,mary,john] been something like *yramlovesnhoj*, where *yram* is a string meaning "mary" when she is the subject of the sentence (as opposed to *m,a,r,y* which is used when she is the object) and *n,h,o,j* is a string meaning "john" when he is the object of the sentence (as opposed to *j,o,h,n* when he is the subject), then the re-

ordered string *n,o,h,j,l,o,v,e,s,y,r,a,m* would be perfectly distinguishable from the string *j,o,h,n,l,o,v,e,s,m,a,r,y* meaning [loves,john,mary]. Thus in this case, a rule for this alternative word order would be created.

Thus it is hoped that, in a situation in which word order can no longer be relied upon for the distinguishing of subject and object syntactic categories, the use of an alternative mechanism, i.e. different subject and object noun categories, might be promoted.

5.2.1 Results

Initially, in the simulations described below, the probability of an utterance being re-ordered was set to the low value of 1%.

The grammars resulting from these conditions (i.e. use of the *probabilistic* parsing and production algorithm in conjunction with "re-ordering" at a probability of 1%), can be subdivided into two types. The first (Type A) has a single syntactic category used to express nouns, and another for verbs. The second (Type B) has two separate noun categories, one of which is used to express the subject of a sentence, and the other for the object.[2] This is in keeping with the pattern we have seen previously. However, what is different here, is the range of different word orders that each type of grammar allows: whilst type B grammars can take full advantage of all the possibilities, type A grammars are more restricted. However, they are not fixed to a single word order, as one might expect. Instead, they tend to express roughly half the space of available possibilities. The set of possible word orders can be subdivided into pairs, where each member of the pair is identical but for

[2]Although two separate noun categories have emerged, and within any given rule one of them is used for the subject of the sentence and one for the object, again there is little consistency *between* rules: a given non-terminal may commonly be used as the subject of a sentence in one of the top-level rules and the object in another.

the fact that the subject and the object have been transposed. If subject and object forms of any given noun are identical, this makes it impossible to determine which of the two rules is being applied and thus distinguish which noun is the subject and which is the object. Therefore, members of these pairs are mutually exclusive in the Type A grammars. Once one member of the pair has already been learnt, if the other is produced by a chance re-ordering event, it will be parsed (incorrectly) by the learner, and thus not added to the grammar.

So, if a sentence is made up of two nouns plus a verb and no other characters, this allows a total of six permutations. Generally, in a Type A grammar, three of these are expressed, without any loss of distinguishability of subject and object. For example, in the case where the word order SOV is allowed, OSV will not be, because this would cause confusion. Other rules such VSO can be perfectly easily distinguished from this however, due to the different positioning of the verb. Essentially, it is possible to divide the set of possible word orders up into three mutually exclusive pairs, SOV-OSV, VSO-VOS and SVO-OVS. The two items in each pair share a common verb position, but have the subject and object inverted: thus grammars with only one noun category used for both subject and object do allow the use of both members of such a pair, as it would not be possible to tell which rule was being applied.

A typical Type A grammar looks like this:[3]

s/[P, X, Y] $\longrightarrow$ [3/Y, 2/P, 3/X]
s/[P, X, Y] $\longrightarrow$ [2/P, 3/X, 3/Y]
s/[P, X, Y] $\longrightarrow$ [3/X, 3/Y, 2/P]

3/john $\longrightarrow$ [q]
3/mary $\longrightarrow$ [t]
3/pete $\longrightarrow$ [u, f]
3/anna $\longrightarrow$ [z, e]

[3]Where P, X and Y are variables over predicates, subjects and objects, respectively.

3/kath $\longrightarrow$ [r]
2/loves $\longrightarrow$ [c]
2/hates $\longrightarrow$ [r, a]
2/adores $\longrightarrow$ [i, t]
2/kisses $\longrightarrow$ [i]
2/sees $\longrightarrow$ [m, j, g]

It can be seen that this grammar exhibits three of the six possible word orders for sentences sentences made up of two nouns, plus one verb: OVS, VSO and SOV. From each of the mutually exclusive pairs SOV-OSV, VSO-VOS and SVO-OVS discussed above, only one member is present. And of the three different word orders that do occur, the position of the verb makes it quite clear which rule is being applied, and thus it is possible to differentiate which noun is subject and which is object.

Occasionally, pairs of mutually exclusive word orderings *are* learnt by individual agents speaking a Type A language, if both are presented very early on in the learning process before any general top-level rules have been induced. Imagine a scenario where the speaker has a Type A grammar, with a basic SVO order. The first meaning it is asked to present to the learner is [loves,john,mary], and it duly selects the string *j,o,h,n,l,o,v,e,s,m,a,r,y*. However, when the second meaning is requested, [loves,anna,pete], a re-ordering event occurs, changing the transmitted string from *a,n,n,a,l,o,v,e,s,p,e,t,e* to *p,e,t,e,l,o,v,e,s,a,n,n,a*. Thus the learner will happily add the rules

s/[loves,john,mary] $\longrightarrow$ j,o,h,n,l,o,v,e,s,m,a,r,y
s/[loves,anna,pete] $\longrightarrow$ p,e,t,e,l,o,v,e,s,a,n,n,a

to its grammar. Later, when the appropriate non-terminal noun and verb categories have been induced, these rules will have become

s/[P, X, Y] ⟶ 1/X, 2/P, 1/Y
s/[P, X, Y] ⟶ 1/Y, 2/P, 1/X

and the agent's grammar *is* actually capable of generating pairs of strings which are identical in appearance, but opposite in meaning in terms of "who did what to whom".

Contrasting the Type A grammar with a typical Type B grammar such as that shown below, we can see that a much wider range of word orders is allowable. This one exhibits four of the possibilities – OVS, SOV, VSO and SVO:

s/[P, X, Y] ⟶ [3/Y, 4/P, 1/X]
s/[P, X, Y] ⟶ [1/X, 3/Y, 4/P]
s/[P, X, Y] ⟶ [4/P, 3/X, 1/Y]
s/[P, X, Y] ⟶ [1/X, 4/P, 3/Y]
1/john ⟶ [i]
1/mary ⟶ [f, z, x]
1/pete ⟶ [h, n, v]
1/anna ⟶ [j]
1/kath ⟶ [y]
3/john ⟶ [a, t]
3/mary ⟶ [d]
3/pete ⟶ [l]
3/anna ⟶ [i, u]
3/kath ⟶ [q]
4/loves ⟶ [c]
4/hates ⟶ [k, h, k]
4/adores ⟶ [h, i, x]
4/kisses ⟶ [f]
4/sees ⟶ [l]

It should be noted that although only four of the possible six word orders are displayed in this grammar, there is nothing to prevent the other two from being added as well. Presumably it is simply the case that the agent has not observed them during its period as a learner. What is important to notice is that this grammar includes both members of one of the pairs of orderings which are mutually exclusive in the Type A grammar: OVS and SVO. This is made possible by the existence of two noun categories. Thus the meaning [loves,john,mary] using the OVS rule would be *dci* which is perfectly distinguishable from [loves,mary,john] using the SVO rule, which is *fzxcat*, even though both sentences involve a word for "mary" followed by a word for "loves" followed by a word for "john".

When re-ordering is added to the simulation, 50.98% of the runs which ran to completion exhibited a Type A grammar, and 43.14% a Type B. 5.88% of runs did not converge on one or the other. This is a significant increase in the occurrence of separate noun categories compared with runs where re-ordering was not used ($p<0.05$), which is itself a highly significant increase ($p<0.01$) in comparison to runs using the deterministic parsing and production algorithm as can be seen from the following table:

	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
deterministic	58.33	19.05	22.62
probabilistic	48.0	33.0	19.0
with 1% re-ordering	50.98	43.14	5.88

Thus it would appear that the attempts to create a selective pressure for some form of case marking have to a limited degree been successful. However, there are some interesting points to note: firstly the introduction of a probabilistic parsing algorithm itself seems to create a mild pressure towards two-noun-category grammars; secondly, the *increase* in the incidence of two-noun-category grammars (+24.09%) is more than three times as large

as the *decrease* in the incidence of one-noun-category grammars (−7.35%) – two-noun category grammars seem to have been largely recruited from the runs that were previously not converging on a minimal compositional grammar; finally, the incidence of non-convergence decreases dramatically when optionality of word order is made possible, going from 22.62% in the original system with the deterministic parser to *5.88%* when re-ordering and the probabilistic parser are used.

5.3 Increasing the amount of word order freedom

If such a small amount of word order freedom (i.e. a 1% chance of re-ordering) is sufficient to increase the frequency of grammars exhibiting case-like behaviour in the form of two distinct noun categories, can we increase their occurrence still further by making the likelihood of re-ordering greater?

Further simulation runs were performed in which re-ordering occurred with a probability of of 2%, 5% and 10%. The results are given below, with those for 1% from the table above included for reference.

probability of re-ordering	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
1%	50.98	43.14	5.88
2%	44.16	46.75	9.09
5%	42.30	48.08	9.62
10%	38.30	38.30	23.40

This appears to show a trend of increasing numbers of Type B grammars relative to Type A grammars as the probability of re-ordering increases, up

to a point. This seems to be coupled with an increase in the number of runs failing to converge on a grammar of either type. However, once the probability of re-ordering reaches the 10% point, this trend ceases, and the proportion of two-noun category grammars seems to drop again. Also, the number of runs failing to converge increases dramatically. However, a chi-squared test on the first three conditions, i.e re-ordering at a probability of 1%, 2% and 5%, shows that despite a generally increasing trend in the proportion of two-noun category grammars, these differences are not statistically significant ($p>0.05$).

Looking at the distribution of nouns in two-noun category grammars, we can see that the degree of consistency with which noun categories are used (i.e. whether a given non-terminal is *always* used to express either the subject or the object of a sentence for a particular grammar) seems to reflect this trend, with the number of grammars using their nouns inconsistently *increasing* as the probability of re-ordering increases, until the condition where re-ordering happens with a probability of 10% is reached, which results in a dramatic *decrease*.

probability of re-ordering	consistently used noun categories	inconsistently used noun categories
1%	75.00	25.00
2%	66.67	33.33
5%	62.11	38.89
10%	88.00	12.00

However, again the differences between the conditions where there is a 1%, 2% or 5% chance of re-ordering do not reach statistical significance ($p>0.05$), although the 10% condition *is* significantly different from the 1% condition.

Increasing the probability of re-ordering also increases the range of possible word orders expressed by Type B grammars – whereas at the 1% level, few

grammars expressed more than 4 of the 6 available possibilities, by the 10% level, most of them express nearly all 6. This result is entirely unsurprising – a higher probability of re-ordering increases the probability that a given word order will be introduced.

A final feature to examine when considering the output from simulations with different probabilities of re-ordering is the proportion of those grammars emerging which are "optimal". Optimality is defined as containing permutations of only a single top level rule, plus appropriate lexical entries for each of the *individuals* and *events* in the meaning space. There is no redundancy: each meaning in the space of possible meanings can be expressed in one way and one way only (with the exception of permutations of word order). The following table shows the percentage of Type A and Type B grammars emerging that can be considered optimal, for different amounts of re-ordering:

probability of re-ordering	Type A (1 noun cat)	Type B (2 noun cats)
1%	61.54	29.55
2%	35.29	19.44
5%	36.29	11.11
10%	40.62	4.00

Two things are immediately apparent: that Type A grammars are much more likely to show optimal characteristics than Type B, even at very low levels of re-ordering, and that in both cases, the chances of an optimal grammar emerging decreases as the chance of re-ordering increases.

Thus it would appear that although freedom of word order in the current system can act as a selective pressure for the emergence of "case-like" grammars, its effectiveness is only limited. The general trend of these results seems to be that a greater degree of re-ordering results in the emergence of a larger pro-

portion of two-noun category grammars relative to the number of one-noun category grammars, and a higher chance that those grammars which have two noun categories will use them inconsistently, coupled with a decrease in the incidence of "optimal" grammars. However, this trend is only weak and does not attain statistical significance. Furthermore, it is only the case for relatively low levels of re-ordering. Once the probability of re-ordering reaches the region of 10% this causes severe disruption to the emergence of minimal grammars, with a much larger proportion of simulations failing to converge on one type or the other.

5.4 Modifying the Bottleneck

As discussed in Sections 5.2 and 5.3, adding a degree of word order freedom to the current paradigm does seem to result in an increase in the frequency with which grammars containing two distinct noun categories, one used to express the subject of a sentence and the other the object, are observed. However, these results are far from dramatic – the number of two-noun-category grammars is still exceeded by the number of one-noun category grammars emerging under all the conditions tested so far, and moreover the differences between the different conditions struggle to attain statistical significance. Furthermore, when Type B grammars do emerge, they are generally far from optimal.

It has already been discussed on page 66 how crucial the presence of a language "bottleneck" is to the emergence of compositional languages in models such as these. Simply put, if an agent cannot hope to sample utterances covering the entire meaning space during its lifetime, this will have the consequence that a language in which the meaning of a string can easily be predicted from the meaning of its parts will stand a much better chance of survival. In the experiments described so far, such a bottleneck is achieved

by virtue of the fact that meanings are drawn from the meaning space at random, *with replacement*, meaning that using a meaning space of 100 possibilities, and requiring agents to produce 100 utterances per generation, it is highly likely that some of the meanings will be used more than once, whilst others will not be used at all. Kirby calculates the probability that a given agent will observe all possible meanings from the entire meaning space during its time as a learner as being $100!/100^{100}$ [43]. Even so, this is clearly not a very tight bottleneck: given that the number of utterances per generation *equals* the size of the meaning space, one would expect that *a large proportion* of the possible meanings will be seen during each generation. Using the equation on page 100, we have been able to calculate that for 100 utterances drawn at random from a meaning space containing 100 items, the estimated coverage is 0.6339. Thus we would expect the average agent to observe approximately 63% percent of the meaning space during its time as a learner.

As discussed in Section 3.3 Brighton [9] has demonstrated that, under appropriate circumstances, the tighter the bottleneck, the greater the stability of compositional languages relative to holistic ones, and thus the greater the degree of selective pressure for the emergence of compositionality. Thus an attempt was made to exploit this finding in conjunction with freedom of word order in the hope that this would result in the more frequent emergence of Type B grammars. Previous experimentation to control the size of the bottleneck by either reducing the number of utterances per generation or increasing the size of the meaning space has proved unfruitful (unpublished data), thus the alternative approach taken by Vogt in [85] was applied. In this approach, the magnitude of the bottleneck is externally controlled by the experimenter by selecting at random a subset of the entire meaning space at the start of each generation, and allowing speakers to produce utterances only for the meanings within this subset. Additionally, in order to fully exploit the statistical effect of the quasi-probabilistic parser, the number of utterances

made per generation was increased from 100 to 1000 to give learners a large pool of utterances to sample from. Thus, if the subset size is set at 50, there is a fairly high likelihood that in the course of 1000 utterances, most of this subset will be observed.[4] Crucially, when a given agent makes the transition from learner to speaker, a new subset of the meaning space is chosen, from which utterances that are presented to the *next* learner will be drawn. It is highly likely that this subset will contain meanings that were not in the subset on which the new speaker was taught. This raises the possibility that the agent may have to present an utterance for a meaning to which it has not previously been exposed. This is important, as the need to creatively produce utterances for hitherto unobserved meanings is an important factor in the emergence of compositional behaviour (see Vogt [86]).

In the first instance, in order to assess the impact of different bottleneck sizes on the model in general, a range of different subset sizes from 10% to 100% of the total meaning space were used, in conjunction with the quasi-probabilistic parsing algorithm but without any re-ordering. The results are shown below:

[4]Again, we can use the equation from page 100 to calculate the coverage: this time the value of N will be 50, and the value of R, 1000, giving a coverage of 0.999999998, which I think we can safely say is approximately 1!

subset size	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
10%	0	0	100
20%	73.68	5.26	21.05
30%	78.95	10.53	10.53
40%	59.10	27.27	13.64
50%	45.00	45.00	10.00
60%	38.10	38.10	23.81
70%	26.10	39.13	34.78
80%	30.43	43.48	26.09
90%	36.36	40.91	22.73
100%	45.00	25.00	30.00

What we can see from these results, is that using a very tight bottleneck, compositionality does not emerge – where only 10% of the meaning space is used by the speaker to instruct the learner, all the simulations fail to converge on a grammar of either Type A or Type B. Instead the grammars that result are almost completely holistic, similar to those seen in the very early generations of the simulations whose results are described in Section 4.2 (see for example the grammar outlined on page 103). Occasional compositional rules do occur, as in that example, but they do not persist. This is largely due to the high probability that the set of utterances presented to a given agent when it is in its learning phase will have very few items in common with the meanings it will be required to express when it enters its adult phase. Thus, if a generalisation resulting in a small amount of compositionality *has* been made whilst the agent is a learner, first and foremost, it is unlikely that when the agent becomes a speaker it will be called upon to produce an utterance that requires those compositional rules in its production at all. Furthermore, even if such an utterance *is* required, it will be insufficient for the same generalisation to be made by the new learner, unless a *second* utterance that also uses the same compositional rule is presented, due to the fact that rules can

only be induced by comparisons *between* utterances. When agents are each using only 10% of the total meaning space, this becomes incredibly unlikely. For this reason, it is generally the case that any compositional innovations that do occur in a given agent's grammar are not even transmitted to the subsequent generation.

However, with a slightly less restrictive bottleneck, the situation changes drastically. With as little as 20% of the meaning space in use, the grammars which emerge switch from holistic to compositional. Compositionality is seen in 100% of cases. This is in some ways not highly surprising, for the learning mechanism employed by the simulation is strongly biased towards compositionality, as pointed out in [85]. However, what is interesting is the way in which this constraint effects the types of grammars seen. Using a bottleneck in which agents are only presented with only 20% of the meaning space when they are learners results in 73.68% of simulations exhibiting a "Type A" grammar, that is to say a grammar possessing a single noun category used to express both subject and object of the sentence. This is a *much higher* proportion than that seen previously (recall from page 133 that when using the deterministic parser, only 58.33% of simulations converge on a grammar with type A characteristics, and that this percentage falls to only 48.00% when the quasi-probabilistic parser is introduced). Furthermore, the Type A grammars that do emerge under the current conditions tend to be much more optimal than previously: the proportion of Type A grammars composed of just 11 rules – one for each of the five *individuals* in the meaning space, one for each of the five *actions* and one top level rule specifying how to combine the strings produced by the other rules (as in the example given on page 108) – is a massive 85.71%. By contrast, with the deterministic parser, using only 100 utterances per generation and no externally imposed bottleneck, as in Chapter 4, only 51.02% of Type A grammars show these optimal characteristics. When the quasi-probabilistic parser is used under the same conditions as in Section 5.1.3, the figure is fairly similar at 56.25%.

Clearly then, this represents a dramatic increase in the proportion of optimal Type A grammars emerging.

However, when using this very tight bottleneck of only 20% of the meaning space, the proportion of Type B grammars emerging falls drastically, from 19.05% with the deterministic parser using only 100 utterances per generation and no imposed bottleneck, and 33.00% when the quasi-probabilistic parser is introduced, to only 5.26% with 1000 utterances per generation but a subset size of 20%. Interestingly, the number of simulations that fail to converge on either type of grammar is similar under all conditions: 22.62% using the deterministic parser, 100 utterances per generation and no externally imposed bottleneck, 19.00% when the quasi-probabilistic parser is introduced, and 21.05% under the present circumstances, with a subset size of 20% but with agents making 1000 utterances per generation.

Relaxing the bottleneck to allow agents to use 30% of the meaning space does not appear to reduce this very strong driving force towards compositionality – if anything the results are slightly better than with the 20% bottleneck: there is a slight rise in the number of simulations converging on Type A grammars, from 73.68% to 78.95%. Of these, 86.67% exhibit optimal characteristics. The number of simulations failing to converge on either a Type A or a Type B grammar falls from 21.05% to 10.53%. There is also a doubling in the number of simulations converging on a Type B grammar, from 5.26% to 10.53%, which is interesting at this stage given the fact that we have not yet introduced any freedom of word order, and therefore that additional pressure for distinguishable subject and object noun categories would not be expected.

Loosening the bottleneck further sees a gradual decrease in the number of Type A grammars, as well as a decrease in the number of them that exhibit optimal characteristics, in conjunction with an increase in the number of simulations failing to converge on either a Type A or a Type B grammar. What is again perhaps a little unexpected is an *increase* in the number of Type B

grammars emerging. This effect seems to be most prominent for bottlenecks where between 50% and 90% of the meaning space are in use: under these circumstances the number of Type B grammars emerging actually *exceeds* the number of Type A grammars. At a subset size of 80%, 43.48% of the grammars emerging are of Type B, which is a very similar result to that obtained with a 1% chance of sentence re-ordering in Section 5.2. Furthermore, although Type B grammars rarely display optimal characteristics, and certainly never with the high frequency that Type A grammars do, the number of optimal Type B grammars does seem to increase notably in this bottleneck range. The following table shows the percentage of grammars of each type exhibiting optimal characteristics for each bottleneck value:

subset size	Type A (1 noun cat)	Type B (2 noun cats)
20%	85.71	0.00
30%	86.67	0.00
40%	69.23	0.00
50%	77.78	11.11
60%	62.50	12.50
70%	66.67	22.22
80%	28.57	10.00
90%	62.50	22.22
100%	66.67	0.00

In conclusion, it seems that when agents are required to make 1000 utterances per generation rather than just 100, the external imposition of a bottleneck is most advantageous in the emergence of compositional behaviour. Furthermore, this effect seems to be most pronounced for very tight bottlenecks: when speakers are allowed to use only 20-30% of the meaning space to converse with learners, there is a very high degree of convergence indeed on optimal Type A grammars. For moderately sized bottlenecks, for example

where between 50% and 90% of the meaning space is available to the learner, the optimality of the Type A grammars decreases, as does the frequency of their emergence, and perhaps slightly unexpectedly, frequency and optimality of Type B grammars emerging *increases*. It seems that a tight bottleneck favours Type A grammars, and a looser one, Type B.

5.4.1 A tighter bottleneck in conjunction with freedom of word order

The next investigation was to determine what effect a degree of word order freedom will have if used in these revised conditions. We have shown in Section 5.4 that the introduction of an externally imposed bottleneck greatly favours the emergence of compositional behaviour. In the absence of word order freedom, this tends to result in large numbers of simulations converging on optimal grammars of Type A. What will happen when we add variability of word order into the mix? Will the putative addition of pressure for distinguishable subjects and objects cause the optimal Type A grammars to be replaced by optimal ones of Type B? And what of the slightly unexpected effect of looser bottlenecks on the emergence of Type B grammars? Will this be reinforced or reduced by the introduction of word order freedom? In order to investigate these questions, the experiments described in Section 5.4 were re-run, again using 1000 utterances per generation, but this time including a 1% probability of re-ordering as described in Section 5.2, the results of which are described below.

Unfortunately, problems occurred with large numbers of simulations failing to complete in a reasonable timeframe. The reason for this was not quite the same as previous occurrence of this phenomenon, seen in Section 5.1.3,

which was due to rules of increasing string length as described by Smith in [70]. Instead it occurred even when there were *just a few* extra characters in the sentence level strings, for example:

s/[P,X,Y] ⟶ 8/X, n, a, 8/Y, a, o, 15/P

This causes an explosion in the number of rules in a grammar, because of the sheer number of combinatorial possibilities: when a grammar contains only 3 non-terminals in its sentence-level rules, there are only 6 possible permutations, thus the use of re-ordering can only potentially add six rules to the grammar. However, if there is a single terminal character in the rule, this already increases the number of permutations to 24. Adding another terminal character increases the number of permutations to 120, and for the example rule given above, where there are 7 items in the rule in total, the number of possible permutations is 5040!

Why is this a problem under the current conditions when it was not when re-ordering was used previously at only 100 utterances per generation? The answer is simply because of the larger number of re-ordering operations that are likely to be made in any given generation: if there is only a 1% probability of a re-ordering event occurring, and only 100 utterances per generation, it is likely that there will only be one or two re-ordering events per generation. However, with 1000 utterances per generation, there could easily be 10 or more in every generation, which will cause a large number of additional rules to be added to the grammar. The severity of this problem seems to increase dramatically when the bottleneck is relaxed: for bottlenecks of 50% and 60%, the number of simulations successfully completing was so few, that it was deemed impractical to run any simulations for the bottleneck sizes beyond this. This is unfortunate, because as previously noted, it is simulations that are in the range where 60% to 90% of the meaning space is in use that seem to result in the largest number of Type B grammars, at least when re-ordering is not present. The table below shows the percentage of simulations that had

to be terminated for a variety of different bottleneck sizes:

subset size	percentage of simulations terminated
20%	30.77
30%	46.15
40%	26.92
50%	57.69
60%	53.85

Of those simulations which did run to completion, the proportion which resulted in grammars of Type A, Type B or which did not converge for each of these bottlenecks are shown below:

subset size	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
20%	55.56	27.78	16.67
30%	50.00	42.86	7.14
40%	42.11	47.37	10.53
50%	54.55	27.27	18.18
60%	58.33	33.33	8.33

It is very difficult to draw any firm conclusions from these results, due to the low numbers of simulations actually completing. However, it does seem possible to tentatively suggest that there has been a move towards the emergence of Type B grammars. For tighter bottleneck values, the numbers of simulations failing to converge on a grammar are very similar with and without the introduction of re-ordering events. However, the distribution between Type A and Type B grammars is quite different: whilst for a bottleneck where the subset size is only 20%, without re-ordering only 5.26% of grammars emerging are of Type B compared to 27.78% when the probability of re-ordering is set at 1%; for a bottleneck of 30%, these figures are 10.53% and 42.86%; for

a bottleneck of 40% they are 27.27% and 47.37%. However, once the bottleneck is relaxed further than this, the proportion of Type B grammars starts to fall again. As when there is no re-ordering, of the Type A grammars that *do* emerge, a very large percentage of them exhibit optimal characterstics when the bottleneck is very tight, but this falls off rapidly as the bottleneck is relaxed (with the exception of the value recorded when a subset size of 60% is used – presumably this is attributable to the low numbers of simulations actually completing). None of the Type B grammars that emerge display optimal characteristics, as can be seen from the following table:

subset size	Type A (1 noun cat)	Type B (2 noun cats)
20%	70.00	0.00
30%	85.71	0.00
40%	50.00	0.00
50%	16.67	0.00
60%	85.71	0.00

To summarise, we can now compare the various conditions we have experimented with to date to determine which of them are most favourable for the emergence of Type B grammars. The following table shows the results from each of the different experimental conditions so far: the original model based on Kirby's [43] and using the deterministic parser, the same model with the quasi-probabilistic parser substituted for Kirby's deterministic one, the effect of a 1% probability of sentence re-ordering and finally the effect of an externally imposed bottleneck added to the probabilistic parser and 1% chance of reordering.

	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
deterministic	58.33	19.05	22.62
probabilistic	48.0	33.0	19.0
with 1% re-ordering	50.98	43.14	5.88
with 1% re-ordering + 40% bottleneck	42.11	47.37	10.53

Clearly from these results it is possible to say that we have been successful in our aim to promote the emergence of grammars with distinguishable subject and object categories, effecting a rise from 19.05% under the conditions of the original model to 47.37% when using the quasi-probabilistic parser, with 1% re-ordering and an imposed bottleneck of 40% of the meaning space. However, it seems that it is not only the presence of alternative word orders that results in this change: as previously noted, simply switching from the deterministic to the quasi-probabilistic parser is sufficient to cause nearly a 14% increase in the frequency of emergence of Type B grammars. Additionally, it has been noted that the introduction of a very loose bottleneck also seems to promote the emergence of Type B grammars, even in the absence of re-ordering, and despite the fact that in other respects it is disruptive to the emergence of compositionality (in that the number of simulations failing to converge on a grammar increases, and the percentage of those grammars which do emerge displaying optimal characteristics decreases). With a very weak bottleneck where 80% of the meaning space is in use, 43.48% of grammars emerging are of Type B – this is actually a very similar figure to that obtained when re-ordering of sentences was first introduced. We will touch upon this point again later on in Section 5.5.

5.5 Discussion

The current work has shown that it is possible for a primitive case system (i.e. separate noun forms to represent subject and object) to emerge from a population of learners equipped with a simple learning algorithm for grammar induction, but no language specific knowledge. When the initial system is changed to incorporate a quasi-probabilistic parsing and production algorithm, as opposed to the original deterministic one, and a degree of word order variability is included, where components of a sentence are re-ordered with a probability of 1%, the number of simulation runs exhibiting such a dual-noun system of case increases from 19.05% to 43.14%. Imposing an external bottleneck which limits agents to only 40% of the meaning space results in a further increase to 47.37%. Thus, the existence of variability in the word ordering of sentences appears to act as a pressure for the emergence of such languages.

The type of language emerging also has an effect on the range of word-orders that a language will *permit*. Where there are two separate noun categories, one for the subject and one for the object, any of the six permutations of subject, object and verb are possible. However, in those populations in which this form of "case" does not emerge, the language need not be restricted to just a single word order, but to a very specific subset of those available: those which are easily distinguishable from each other by the positioning of other elements of the utterance.

This is a little strange because such a pattern of restricted word order is quite unlike natural language, where it is generally the case that word order is either fairly strict, as in English, or allows a full range of possibilities, e.g. languages such as Turkish and Serbo-Croatian [69]. Even when languages do exhibit a restricted set of word orders, such as Italian, which is predominantly SVO, but in which OVS, VSO and VOS are also relatively

commonplace, this subset often includes pairs of word orders for which other cues are required to specify subject and object. Interestingly, Italian does not have a particularly rich case structure, and these alternative word orders are generally only allowed when the meaning can be disambiguated from the context. Of course, in natural language, mechanisms other than the need to disambiguate are at work in the development of case. Otherwise there would be no reason why languages should exhibit completely fixed word order, and patterns such as that found in Italian would be somewhat unlikely. This may perhaps be related to sentence processing demands: as mentioned in Chapter 3, Lupyan and Christiansen [53] have done studies using simple recurrent networks, which demonstrate that certain word orders are easier to learn than others. In particular, SVO and VSO are more readily acquired than SOV, but the addition of case markings aids acquisition of the latter. Therefore it is possible that case markings may originally have appeared in order to facilitate the acquisition of fixed word order languages whose underlying word order is difficult to learn, but that their existence might enable more freedom in the word order used. These considerations are clearly not an issue in the current system, as all possible word orders are equally easy to learn and equally likely.

So, the introduction of word order freedom into the current system does appear to create a selective pressure for the emergence of case. However, it would appear that this is not a very *strong* pressure: Firstly, whilst we are able to increase the proportion of grammars emerging with two noun-categories, we still only just manage to exceed the number of grammars with only a single noun category, meaning that this type of grammar is still perfectly viable, even in the face of variable word order. Secondly, "case" is promoted simply by the use of a probabilistic rather than a deterministic algorithm for sentence generation, by mechanism(s) unclear, and also seems to be favoured by the imposition of a weak external bottleneck of the order that would normally be detrimental to the emergence of compositionality. Fur-

thermore, increasing the amount of word order freedom does not significantly increase the emergence of two-noun-category grammars. Even the existence in a grammar of *both members* of a supposedly "mutually exclusive pair" of word orders is not sufficient to guarantee the emergence of such a grammar. The proposed mechanism by which adding word order freedom was hoped to promote the emergence of case was in order to disambiguate potentially conflicting word orders. However, the system as it currently stands does not include any requirement for agents to *understand* each other, so in reality, there *is* no need for disambiguation: if the re-ordering produces an utterance that can be confused with a meaning other than that intended, rules for that word ordering are simply not acquired. There is nothing within the system that requires the speaker to make itself understood or the learner to be able to understand. It is simply a case of whether utterances can be parsed or not – if they can, then no action is taken, even if the wrong meaning is returned by that parse, and if they cannot then learning occurs. This means that if a grammar currently has just a single noun category, and if by invocation of the re-ordering procedure, the speaker happens to transpose the subject and object, the learner will just assume that the speaker meant something other than what was truly intended. There will be no pressure on speakers to express themselves in a way that learners will not misinterpret, such as using different noun forms for each *individual* to indicate whether it is subject or object of the sentence.

Thus it seems that the need to disambiguate conflicting word orders is not a factor at play here in the emergence of Type B grammars. In that case, why are they favoured by the introduction of re-ordering? Recall that in the end of Section 5.4.1 that we mentioned that the emergence of Type B grammars is promoted simply by introduction of the quasi-probabilistic parsing algorithm, and again further when an external bottleneck is imposed, but only if this is a relatively weak one which in all other respects would be disruptive to the emergence of compositionality. So perhaps it is the case

that Type B grammars are in fact the result of some degree of breakdown in compositionality, that occur when conditions make it difficult for a truly compositional grammar to emerge? This would certainly appear to be true of the first two conditions in which their emergence is promoted: the quasi-probabilistic parser represents a breakdown between the one-to-one mapping of meanings and strings, which Smith [70] has shown is so crucial to the emergence of compositional behaviour, and the size of bottlenecks at which Type B grammars start to predominate are the more relaxed bottlenecks at which compositionality is less favoured. So perhaps in the case of freedom of word order, rather than being the introduction of potentially ambiguous utterances at play in the emergence of Type B grammars, it may just be the disruptive effect of re-ordering sentences making it more difficult for agents to converge on a compositional grammar and thus resulting in a suboptimal solution. In Chapter 6 we will investigate this hypothesis further and see whether the introduction of measures that require agents to resolve ambiguous utterances makes a difference to these results.

The other notable point about the results achieved thus far is that the "case system" that emerges in the current study is far from representative of the type of case found in natural languages, which normally takes the form of inflectional affixes. In the present results, subject and object forms of the same noun are completely unrelated. Whilst this might occur for certain irregular word-forms that are very frequently used in a given language (such as "we" and "us"), it is not the norm. In Chapter 7 we will attempt to extend the current system to generalise across any chance regularities that may occur between subject and object forms of a given noun, or across different nouns of the same case, in the hope that a truly inflectional case system may be derived.

5.6 Summary

In this chapter, we have described modifications to Kirby's Iterated Learning Model which enable agents to acquire and productively use multiple word orders. We have then introduced a degree of word order variability into the system to investigate what effect this has, if any, on the type of grammars emerging. It was discovered that this does indeed result in an increase in the number of "Type B" grammars, i.e. those with two distinct noun categories, relative to the number of grammars of "Type A". The imposition of an external bottleneck on the proportion of the meaning space to which agents are exposed as learners also facilitates the emergence of "Type B" grammars under certain conditions. However, due to the fact that there is no requirement for agents to *understand* each other, the pressure imposed by the existence of multiple word orders does not seem to be as strong as it might be. In Chapter 6 we will attempt to address this by making changes to the system whereby the speaker *is* required to make itself understood, in the hope that this will cause a further increase in the incidence of case-like behaviour.

Chapter 6

Rejecting Ambiguous Utterances

In Chapter 5 we demonstrated how the occasional re-ordering of utterances spoken by an agent can be used to create a selective pressure for the emergence of grammars exhibiting a primitive form of case. By this we mean grammars in which the rules determining the structure of a sentence use two distinct noun categories, one to express the subject, and the other for the object. We have termed a grammar of this type a "Type B" grammar, and noted that it allows complete freedom of word order: rules for any of the possible permutations of subject, object and verb order may be added to the grammar if such an utterance is heard. Conversely, for a "Type A" grammar, in which the rules for constructing a sentence all contain instances of only a single noun category used to express *both* subject *and* object, different word orders *are* permissible, but they are constrained to those that can be easily distinguished from one another by the position of the non-noun elements (i.e. the verb, and any top level terminal characters).

It was also discussed how the pressure created by this intervention appeared

to be a relatively weak one, as the presence of re-ordered sentences in the languages being spoken by agents during the course of a simulation did not seem to guarantee the emergence of such case-like behaviour. Although their introduction certainly does result in an increase, it is noted that simply changing the parsing algorithm used by agents in the simulation so that it is quasi-probabilistic rather than deterministic is itself sufficient to promote the behaviour being sought to a certain degree. Further to this, increasing the frequency with which utterances are re-ordered, above and beyond the initial setting of a 1% probability does not produce a significant further increase. And finally, experiments with external manipulation of the transmission bottleneck showed that tight bottlenecks favouring the emergence compositional behaviour seem to swing the output of simulations almost totally in favour of Type A grammars, even in the presence of re-ordered sentences. However, looser bottlenecks, which are otherwise detrimental to the emergence of compositionality, do seem to favour Type B grammars. Unfortunately though, the absolute proportion of such grammars is only slightly greater than under circumstances where the bottleneck is not manipulated.

6.1 Ensuring the Presence of Ambiguous Word Orderings

The logic behind the introduction of occasional re-ordering of sentences is that this will create a pressure for Type B grammars by introducing ambiguity. When a chance re-ordering event occurs, the learner that witnesses this event will add rules to its grammar for parsing and creating sentences with that alternative word order. Under certain circumstances, this might lead to utterances where it is not possible to determine which noun was intended as the subject of the sentence and which the object, unless each noun exists in two different forms.

However, the vast majority of "re-order" operations do not result in any ambiguity at all – as already noted, even Type A grammars tolerate some freedom of word order: several rules for constructing sentences with the words in different orders are perfectly permissible as long as they can easily be distinguished from each other. If the hearer can tell with ease which rule has been applied, then it will have no trouble interpreting the sentence correctly. Therefore, as long as any rule induced from a re-ordered utterance can be distinguished from the other rules currently in the grammar, no ambiguity results. There is only one circumstance under which ambiguity *is* the outcome, in fact: when the word order observed is exactly the same as one already encompassed by the agent's grammar, except with the subject and object inverted. Obviously, the greater the range of word orders a grammar already contains, the higher the probability that a re-ordering event will result in such a rule, but it still remains the case that regardless of whether a grammar is of Type A or Type B, 50% of the possible word orders can coexist in it without resulting in any ambiguity at all. Thus their introduction by a re-ordering event is not not likely to result in a selective pressure for the emergence of case. If an agent has not yet learnt *any* alternative word orders, then re-ordering will result in ambiguity in very few cases: take for example a sentence rule which contains just two nouns and a single verb, but no other characters, such as top-level terminals. This means that there are five alternative permutations of the elements of the sentence which could be created by a re-ordering event: only one of these will result in ambiguity – that in which the subject and object have been transposed.

In order to try and increase the pressure for Type B grammars that might be generated by re-ordering, experiments were carried out in which it was *guaranteed* that it will result in ambiguity. This was achieved by creating an "invert" procedure, to be used in the place of the "re-order" previously described, which identifies the substrings associated with the subject and ob-

ject of the sentence, if such substrings exist,[1] and swaps their positions. And rather than invoking this operation with a fixed probability, p, instead an absolute number of utterances to which it will be applied in each generation is specified. Thus, during its period as learner, each agent can be *guaranteed* to hear a certain number of utterances in which the meaning of the string according to the prevailing grammar is the opposite of what it appears to be. The experiments described in Section 5.2 were repeated, with agents using the quasi-probabilistic parser, and making 100 utterances per generation, but rather than invoking the "re-order" operator with a probability of 1%, they have the "invert" procedure applied to a fixed number of their utterances. In the simulations described below, this was set to one utterance per generation in order to give some degree of comparability between the experiments described in Chapter 5 and the current ones.

Thus it is hoped that by focusing on those transformations that will produce utterances that are ambiguous, then if the re-ordering introduced in Chapter 5 is indeed having the anticipated effect, (i.e. creating a selective pressure for the emergence of Type B grammars, which use two distinct noun categories to express subject and object of a sentence), this intervention should result in an even further increase in the behaviour being sought.

6.1.1 Unexpected results

However, this would appear to be far from the case. What is apparent immediately is the extent to which this simple step causes a significant decrease in the probability of convergence on a grammar of *either* type. Not only is there an increase in the number of simulations that need to be terminated (usually due to the accumulation of multiple very long rules containing many

[1] In rules where either subject or object have not yet been lexicalised, i.e. those rules which are still holistic in part, no action is taken.

repeated segments, similar to those described in Sections 5.1.1 and 5.1.2), but also the number of runs which fail to converge according to our definition (i.e. those which do not arrive at a consensus regarding the categories being used to make up a sentence) also soars dramatically. These runs make up only 5.88% of the total under the conditions described in Section 5.2, where there is a 1% chance of re-ordering for each utterance, but rise to 44.62% where inversion is guaranteed in 1 utterance per generation. Furthermore, of those simulations which *do* converge, far from exhibiting an increase in the proportion of Type B grammars emerging, they actually appear to show a significant decrease. When there is a 1% chance of re-ordering, the number of grammars exhibiting two noun categories is roughly equal to the number of those with only one, but when one utterance per generation has had its subject and object inverted, there are only half as many two-category grammars as one-category ones. The following table shows the percentages of each grammar type emerging under each condition:

	Type A (1 noun category)	Type B (2 noun categories)	non-converging
with 1% probability of re-ordering per utterance	50.98	43.14	5.88
with inversion of subject and object (one utterance per generation)	36.92	18.46	44.62

6.1.2 Why is this so?

So why does the guaranteed presence of utterances in which the subject and object have been inverted not promote the emergence of Type B grammars? And why, in fact, does it result in a decrease in their emergence? As discussed previously, it was hoped that the introduction of re-ordered sentences in general, and those where subject and object have been transposed in particular, would promote case-like behaviour by creating a need to disambiguate potentially conflicting word orders. Why is this not occuring?

The failure seems to be due to the fact that the current system does not include any requirement for agents to *understand* each other. Thus, such disambiguation is unnecessary: an agent hearing an utterance which can be parsed by its grammar, regardless of the meaning returned, will accept that utterance quite happily and not take any further action. If the meaning returned by the parse is incorrect, this will have no bearing whatsoever; the hearer will not make known to the agent producing the string that the message has been misinterpreted, nor will it attempt to add the string in question to its own grammar in association with the intended meaning in any way. Thus there is no pressure on speakers to produce non-ambiguous utterances.

Furthermore, because the *alternative* meaning for the string in question will not be added to the agent's grammar, it will not be propagated from one generation to the next. Such ambiguities can only exist when they arise *de novo*, from a re-ordering event for example. Thus, rather than *promote* the emergence of grammars with two distinct noun categories, pairs of rules allowing subject and object to occupy interchangeable positions in a sentence can in fact *only* exist in such grammars. In grammars with only a single noun category, they are lost as soon as they appear, and there is no pressure for the emergence of second category.

To put it simply: the presence of different word orders in a grammar does not have any effect on the number of noun categories a grammar contains, but rather the number of noun categories in existence *constrains* the range of word orders that will be acquired.

Effectively then, the introduction of word order freedom into the system as it is currently will result in very little pressure for the emergence of two separate noun categories. It seems that the effects observed in Chapter 5 were thus brought about by some other mechanism. It would seem likely that this is related to the fact that merely changing the parsing algorithm that agents employ from deterministic to probabilistic also causes an increase in Type B behaviour.

In many respects, Type B grammars can be considered *suboptimal* – they are a failure of a simulation to converge on a tidy minimal grammar with which it may express the meaning space required. A non-deterministic parsing algorithm is disruptive. It destroys the one to one correspondence between signal and meaning that Smith [70] found to be so crucial for the emergence of language-like behaviour. Thus it is perhaps no surprise that switching from a deterministic parser to the quasi-probabilistic one described in Section 5.1.3 results in an increase in the sub-optimal behaviour that we refer to as *Type B grammars.*

Presumably then, the introduction of word order freedom, as another disruptive influence, is propagating this behaviour further for a similar reason.

6.2 A Learner That Does Not Tolerate Ambiguity

Attempts were made to create pressure for the emergence of Type B grammars in the presence of word order freedom by changing the model to *select against* grammars that contain ambiguities. The most obvious strategy seemed to be to have the learner reject any string that can be parsed by its grammar to yield a meaning other than that intended by the speaker, thus forcing speakers to produce unambiguous utterances. When a string is rejected, the speaker must find an alternative way of expressing that meaning, and the counts associated with the rules that produced the original string are not incremented. Thus the experiments performed in Section 5.2 were repeated once again, with agents using the quasi-probabilistic parser, producing 100 utterances per generation and with 1% probability of re-ordering occurring. However, when parsing utterances, learners compare the result of that parse with the intended meaning and if it is incorrect, the speaker is requested to produce another utterance for that meaning. The following table shows the percentage of simulations resulting in Type A grammars, Type B grammars, and also those failing to converge on either under these conditions. The results previously obtained in Section 5.2, where ambiguous utterances are not rejected, are also included for comparison.

	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
ambiguity tolerated	50.98	43.14	5.88
ambiguity rejected	40.84	22.54	36.62

Clearly then, this intervention is having a destabilising effect: the number of simulations failing to converge on a grammar has increased more than six-fold, and there has been a noteable decrease in both the number of Type

A and Type B grammars. Interestingly, Type B grammars seem to have suffered the greater losses, despite their being the type of behaviour that these changes were intended to promote.

There is also a decrease in the number of optimal grammars emerging (i.e. those with a single top-level rule for creating sentences, plus one rule for each of of the *events* and *individuals* in the meaning space, or in the case of Type B grammars, one rule for each of the *events* and two for each of the textslindividuals):

	Type A (1 noun cat)	Type B (2 noun cats)
ambiguity tolerated	61.54	29.55
ambiguity rejected	13.79	0.00

And of those grammars containing two noun categories, the chances of each of them being used consistently to express either subject or object of the sentence in all the rules of a given grammar is hugely decreased, from 75% in the former case to 18.75% in the latter.

Figure 6.1 shows the relative proportions of Type A and Type B grammars under the two different conditions (ambiguity tolerated vs ambiguity rejected). What is clear immediately from this diagram is that, whilst the split between the two is approximately 50:50 when ambiguity is tolerated, there is a slight change in favour of *Type A* grammars when agents reject any string that, according to the prevailing grammar, yields a meaning other than that intended. Thus it seems that rejecting utterances which may be interpreted ambiguously is actually having the opposite effect to that intended.

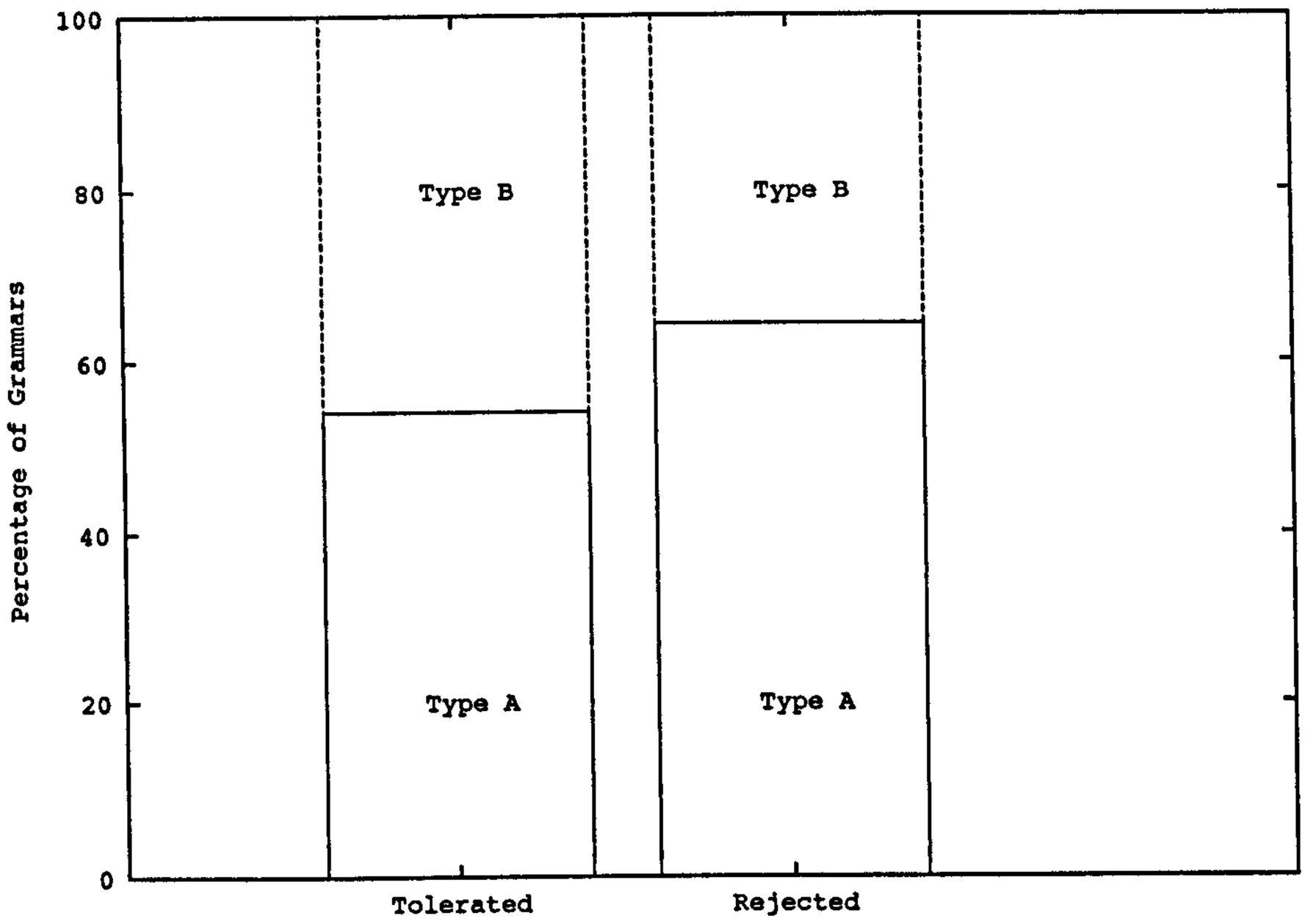

Figure 6.1: *The relative proportions of simulations resulting in Type A and Type B grammars under conditions where a) ambiguity is tolerated and b) ambiguity is rejected by the learner, and the speaker is prompted to produce an alternative string for the required meaning. It can be seen that rejecting ambiguity causes a slight swing towards Type A grammars.*

6.2.1 An increase in redundancy

How might be the decreased stability amongst the grammars emerging from these simulations be explained? It seems to be because the changes we have made promote and prolong the existence of redundancy. When an utterance is produced by the speaker which is wrongly interpreted by the hearer (perhaps caused by the invocation of the re-order procedure resulting in transposition of subject and object in a rule using the same noun category for both) it is rejected. This causes the speaker to scan its grammar for another way of saying the same thing (more specifically, perhaps a rule which uses different noun categories to express subject and object). However, if there

is *no* other way of producing that meaning, it is forced to invent a string for it, thus adding new redundancy to the grammar. This of course means that new strings are constantly being introduced to the grammar when they would not otherwise be. And if another utterance *does* exist, this will have the effect of augmenting the counts on the rules used to produce this second utterance when they would otherwise not be augmented. This would be fine if the first utterance (and thus the rules used to produce it) were *always* likely to be misinterpreted by the hearer, and thus always rejected in favour of the second utterance, but in a situation such as the hypothetical one described above, this has only occurred due to the chance inversion of the subject and object in a re-ordering event. Thus, in the majority of instances where the rules used to create this particular utterance are used, the hearer will be able to correctly interpret the string without difficulty. As a result, forcing the speaker to produce a new utterance for strings which are misinterpreted merely has the effect of introducing and prolonging the existence of multiple ways of producing a string for a given meaning, slowing convergence, and in some cases preventing it completely due to a breakdown of the one-to-one correspondance between meaning and strings that Smith has shown to be so crucial.

6.2.2 Greater intolerance by means of a penalty

In an attempt to overcome this, a penalty was applied to each of the rules that the speaker used to create the utterance that was misinterpreted by the hearer. This was done by *decrementing* the counts on those rules by a fixed amount. This amount is specified parametrically. In the simple case, it is just 1. The table below shows the number of simulations that converge on Type A grammars, Type B grammars, or that fail to converge on a grammar of either type when this penalty is applied. The results of the experiment performed in Section 6.2, where ambiguous utterances are rejected but no

penalty is applied, are included for the purposes of comparison.

	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
no penalty	40.84	22.54	36.62
penalty	43.24	16.21	40.54

It appears that the introduction of a penalty does not have much of an impact – the percentage of simulations that fail to converge remains fairly similar and the number of the grammars containing two noun categories as opposed to just a single one actually shows a further decrease. Clearly then, penalising the rules which have produced these ambiguous utterances is not having the desired effect (i.e. creating a pressure for the emergence of distinguishable subject and object categories). Could this be because the penalties are not harsh enough? Decrementing the counts of the offending rules by just 1 is probably not sufficient to cause them to be removed from the grammar altogether, but may serve to reduce their dominance somewhat. This might have the result of further encouraging the redundancy discussed above. Therefore this experiment was repeated with a range of penalties of different sizes to investigate the impact of this.

Figure 6.2 shows the number of simulation runs converging on either type of grammar, as a percentage of runs completed, for a variety of different penalties.

This does not appear to be very promising – an increased penalty size does not appear to correlate with a decrease in the number of simulations failing to converge on either type of grammar. In fact, quite the opposite – although there is a *slight* increase in the number of simulations converging for a small penalty (the optimum appears to be around 2), the overall trend is downward, especially for very *large* penalties (although not at a statistically significant level). Generally, the percentage of simulations converging seems to fluctuate

Figure 6.2: *The number of simulations resulting in a converged grammar, either of Type A or Type B, for a range of different penalty values, as a percentage of runs completed.*

around the level of 60%, which is far inferior to the degree of convergence seen when ambiguity is tolerated, but other conditions are similar (that is, using the quasi-probabilistic grammar, and with re-ordering at a probability of 1%), where it occurs in 94.12% of cases.

Turning our attention to the behaviour of those simulations which *do* converge on a grammar, Figure 6.3 shows the relative percentages of simulations resulting in a grammar containing just single noun category used to express both subject and object of the sentence (Type A), and those with *two* distinct non-terminal categories, of which, for any given sentence, one is used to represent the subject and the other the object (Type B), respectively, for a range of different penalty values.

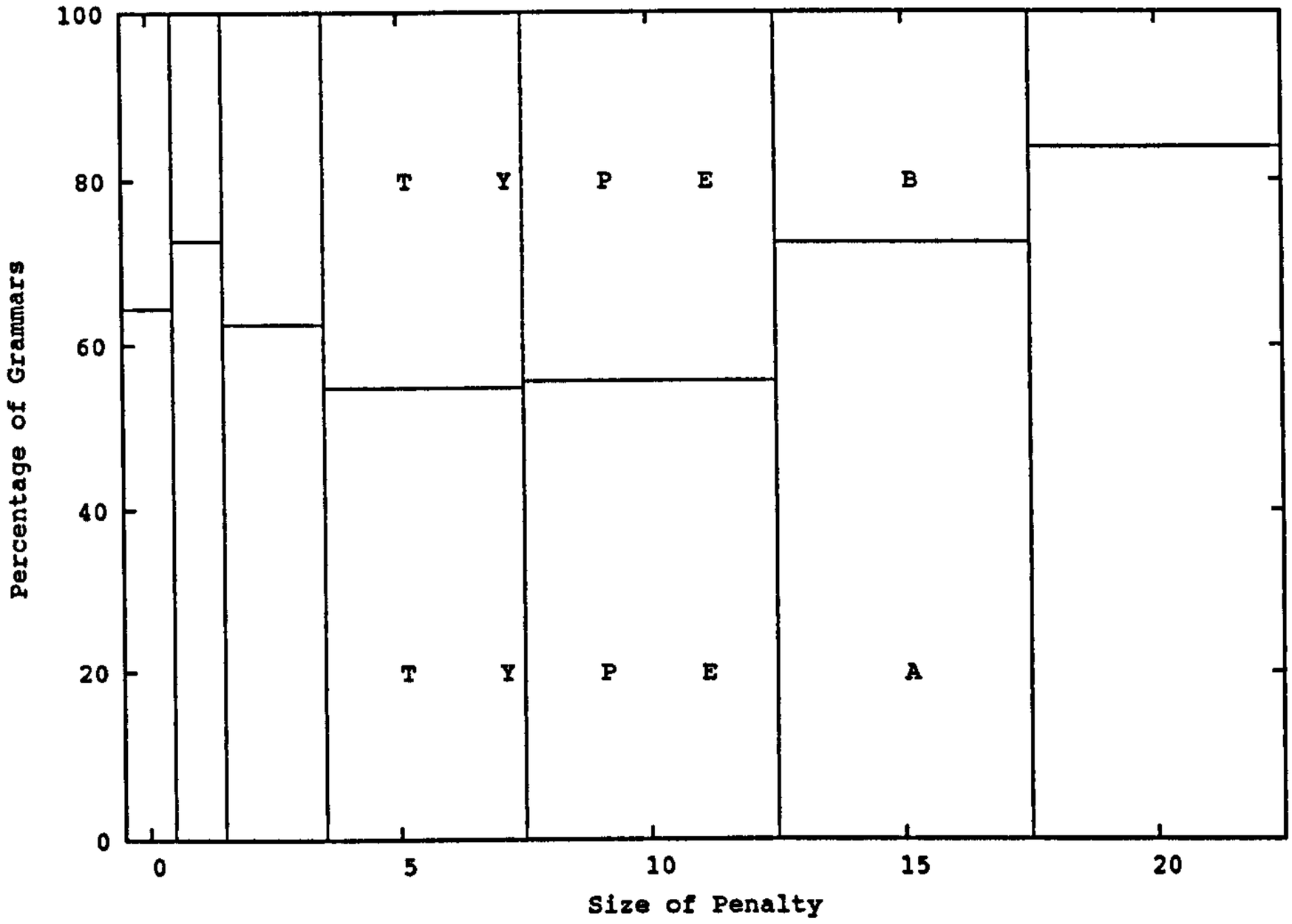

Figure 6.3: *The relative proportions of runs resulting in Type A and Type B grammars as a percentage of those simulations which do converge on a grammar.*

It would appear that, although the introduction of a penalty in the first place appears to cause a slight increase in the number of Type A grammars, when the value of that penalty increased there is a slight swing towards simulation runs exhibiting Type B behaviour. At least for moderate values. With further increases in the size of the penalty, the swing is back towards single-noun category grammars. By the time a value of 20 is reached, (the highest tested here) the vast majority of those simulations which do converge on a grammar are of Type A. The optimal penalty value seems to be somewhere between 5 and 10.

Disappointingly though, even for those runs using the optimum penalty, the proportion of simulations converging on a Type B grammar never even reaches 50%, and is at its very best only similar to that achieved when am-

Figure 6.4: *The number of simulations resulting in Type A grammars for a range of different penalty values, as a percentage of the total number of runs completed.*

biguity is not rejected at all, merely ignored.

Looking further at the numbers of Type A and Type B grammars emerging as a proportion of *all* completed simulations (as opposed to just those which converge) as in figures 6.4 and 6.5, we can see that the numbers of both types of grammar show an overall decreasing tendency. What is noteworthy though, is what happens in the region of the optimal penalty value mentioned before (between 5 and 10): whilst the number of Type B grammars do not appear to be increasing greatly in this range, there *is* a slight decrease in the number of Type A grammars.

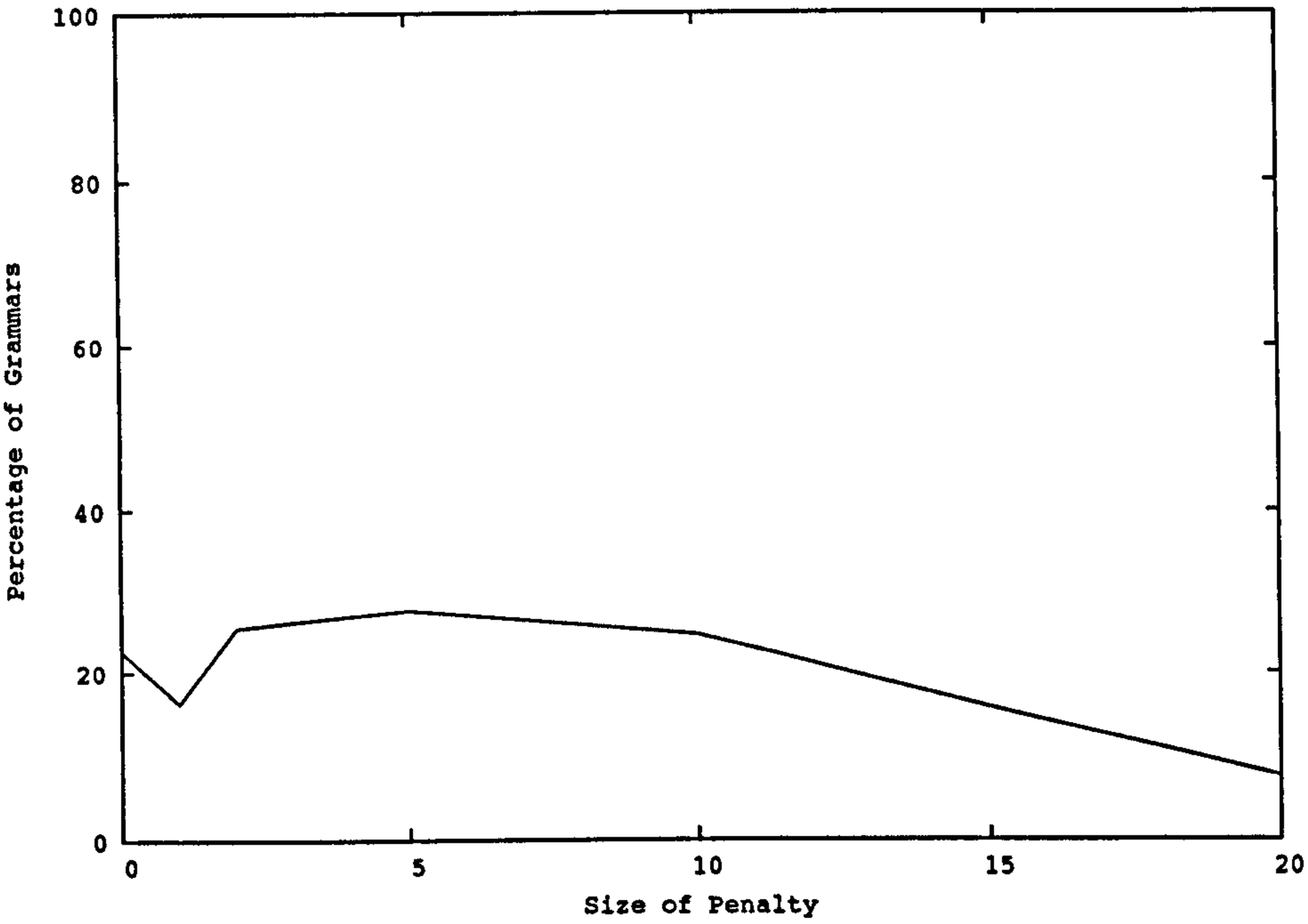

Figure 6.5: *The number of simulations resulting in Type B grammars for a range of different penalty values, as a percentage of the total number of runs completed.*

6.2.3 Reclassifying the output of the simulations

So it would appear that neither rejecting ambiguity nor the introduction of a penalty for rules that produce ambiguous utterances is effective in creating a selective pressure for the emergence of case. Instead these measures seem to cause a great increase in the number of simulations that do not converge on a compositional grammar of either Type A or Type B. In particular, in the range of the optimal penalty value of 5 to 10, there seems to be a slight move away from Type A grammars, but this is not reflected in the number of Type B grammars emerging, and instead, a greater number of simulations fail to converge on either. But what of these unconverged grammars? Can examining them shed any light on the situation?

The tale they have to tell is fairly interesting, because many of them only contain slight irregularities, meaning that it is still possible to classify them as exhibiting either Type A or Type B characteristics. To elaborate, our definition of convergence states that of the top level rules in a grammar, all those that are fully lexicalised should show consistency in the non-terminal category they use for the different parts of speech. So, for example, if category 3 is used to express the verb in one top-level sentence rule of a grammar, it must be used in all of those which contain a lexicalised verb category or the grammar will be considered not to have converged. Similarly, if there are any terminal categories at this level, then the same terminal categories must appear in all the fully lexicalised top-level rules of the grammar. However, in the case of many of the non-converged grammars resulting from simulations where ambiguity has been rejected, there are often a number of fully lexicalised rules that contain the same non-terminal categories but differ in the number and/or type of any additional terminals, as is the case for the two top-level rules in the example given below:

s/[P,X,Y] ⟶ 5/X,x,2/P,3/Y
s/[P,X,Y] ⟶ 5/X,x,n,c,2/P,3/Y
2/sees ⟶ d, i
2/kisses ⟶ p
2/hates ⟶ o
2/loves ⟶ i, u, g
2/adores ⟶ k
3/mary ⟶ r
3/anna ⟶ x, n, c, a, x
3/john ⟶ a, x, k, n, y, e
3/kath ⟶ a
3/pete ⟶ s, d
5/pete ⟶ d
5/kath ⟶ x, n, c, s, g
5/anna ⟶ r, x, d
5/john ⟶ t
5/mary ⟶ n, y, e, g

Furthermore, even in those grammars in which all the lexicalised top level rules do not contain the same non-terminals, there is still often consistency over whether there is a single non-terminal used to express both subject and object of the sentence, or two. Thus, in the simple case where ambiguity is rejected but no penalty is applied, 29 of the 78 simulations run converged on a grammar of Type A, with only a single noun category, and 16 resulted in a Type B grammar, with two noun categories. 26 simulations resulted in did not converge on either type of grammar, according to the definition above. However, of these 26, 10 are grammars which have the same three non-terminal categories in all fully lexicalised toplevel rules, and merely differ in the additional terminal symbols found in them. A further 3 show consistency in the *type* of the top level rules, even if the categories used are not the same. The remaining 13 are mixed – they contain top level rules of both types. Thus it is possible to go back over the results of the previous simulations,

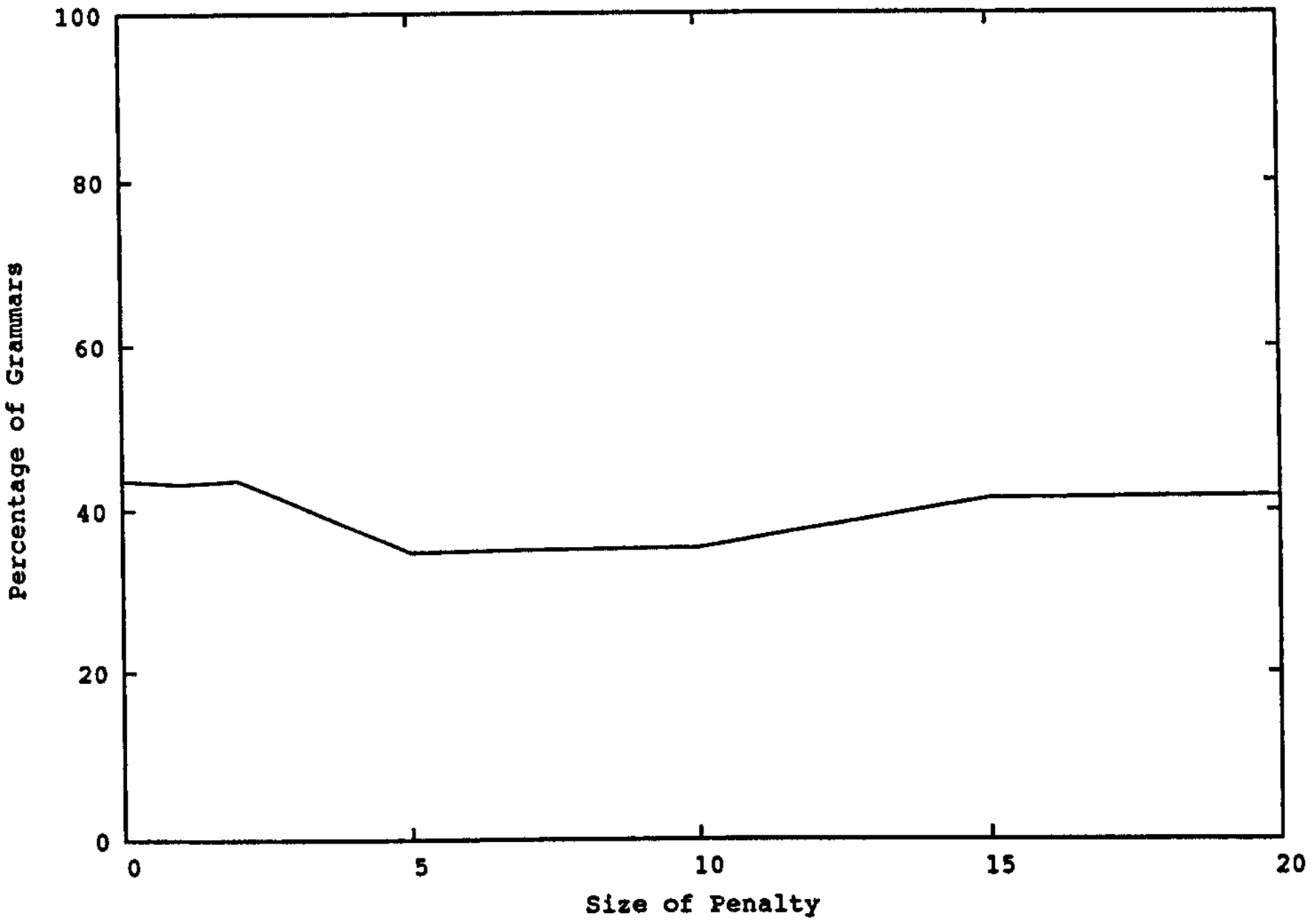

Figure 6.6: *The percentage of simulations resulting in grammars containing only Type A rules for a variety of different penalty values.*

and reclassify them: we shall consider any grammar that contains only Type A rules (i.e. those containing a single noun category used for both subject and object of the sentence) to be of Type A, nd similarly, any that only contain Type B rules (with two distinct non-terminal categories for nouns, one used for the subject of the sentence and the other for the object) will be considered to be of Type B. Any grammar that contains rules of both Type A and Type B will be classified as "mixed".

The results of this re-classification are displayed in figures 6.6 to 6.8, which show how the numbers of Type A, Type B and mixed grammars emerging vary with the size of the penalty.

These results appear to show that the number of simulations resulting in grammars with only Type A rules does not vary much as the size of the

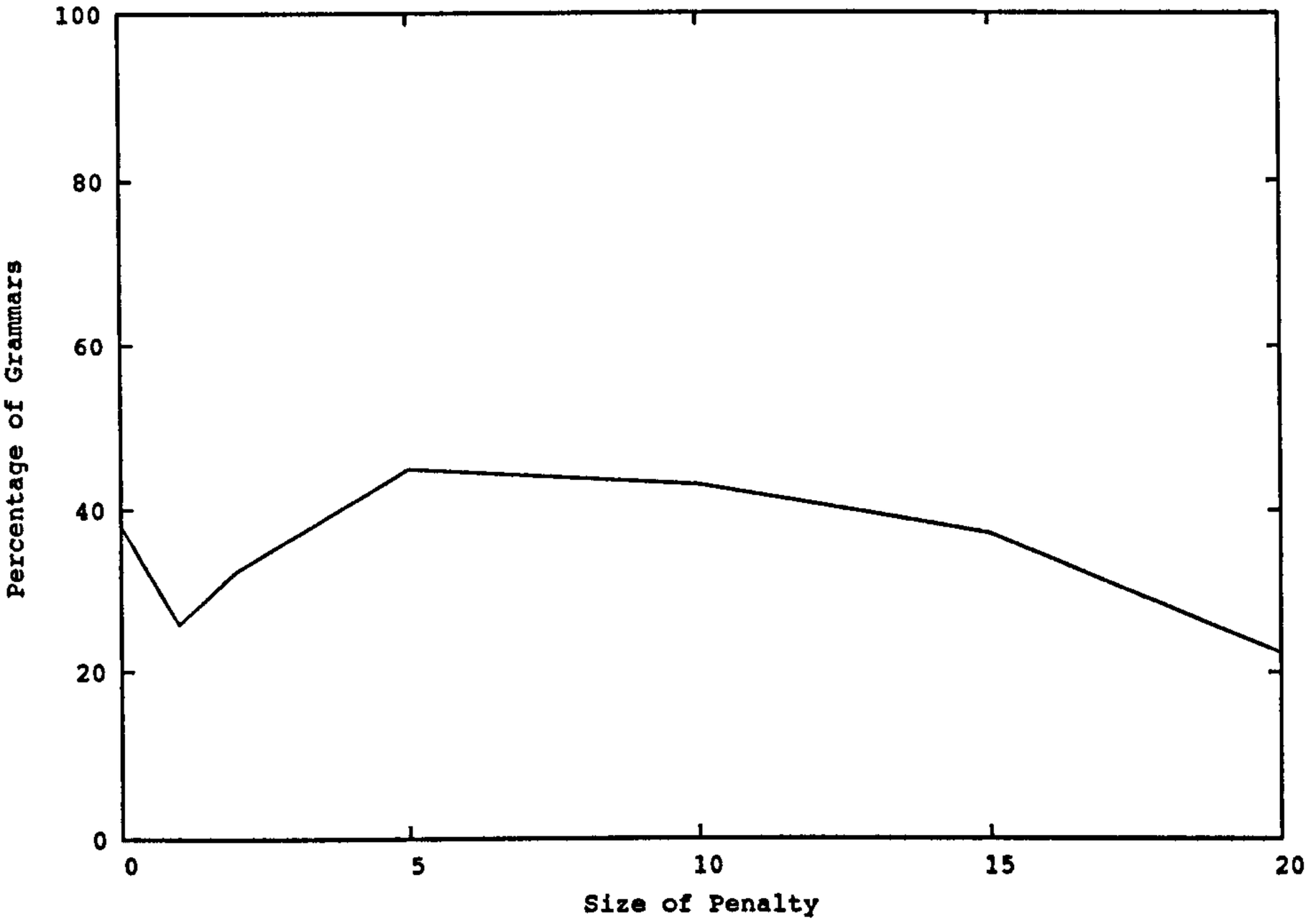

Figure 6.7: *The percentage of simulations resulting in grammars containing only Type B rules for a variety of different penalty values.*

penalty changes, although it does decrease a little for middling penalty levels, being at its lowest for values of between 5 and 10. By contrast, the initial introduction of a penalty appears to cause a slight decrease in the number of simulations resulting in grammars containing only Type B rules, but after this, there is a definite climb in their numbers as the penalty value increases, reaching a maximal level at about 5, beyond which the trend begins to decrease again. This is mirrored in a slight increase in the number of mixed grammars when the penalty is first introduced, followed by a decrease towards a the middling penalty values, and then a further increase for very large penalty sizes.

So, in summary, it seems that introducing a penalty in the first place causes a reduction in the number of simulations resulting in grammars containing only Type B rules, but as the penalty size increases, these start to increase

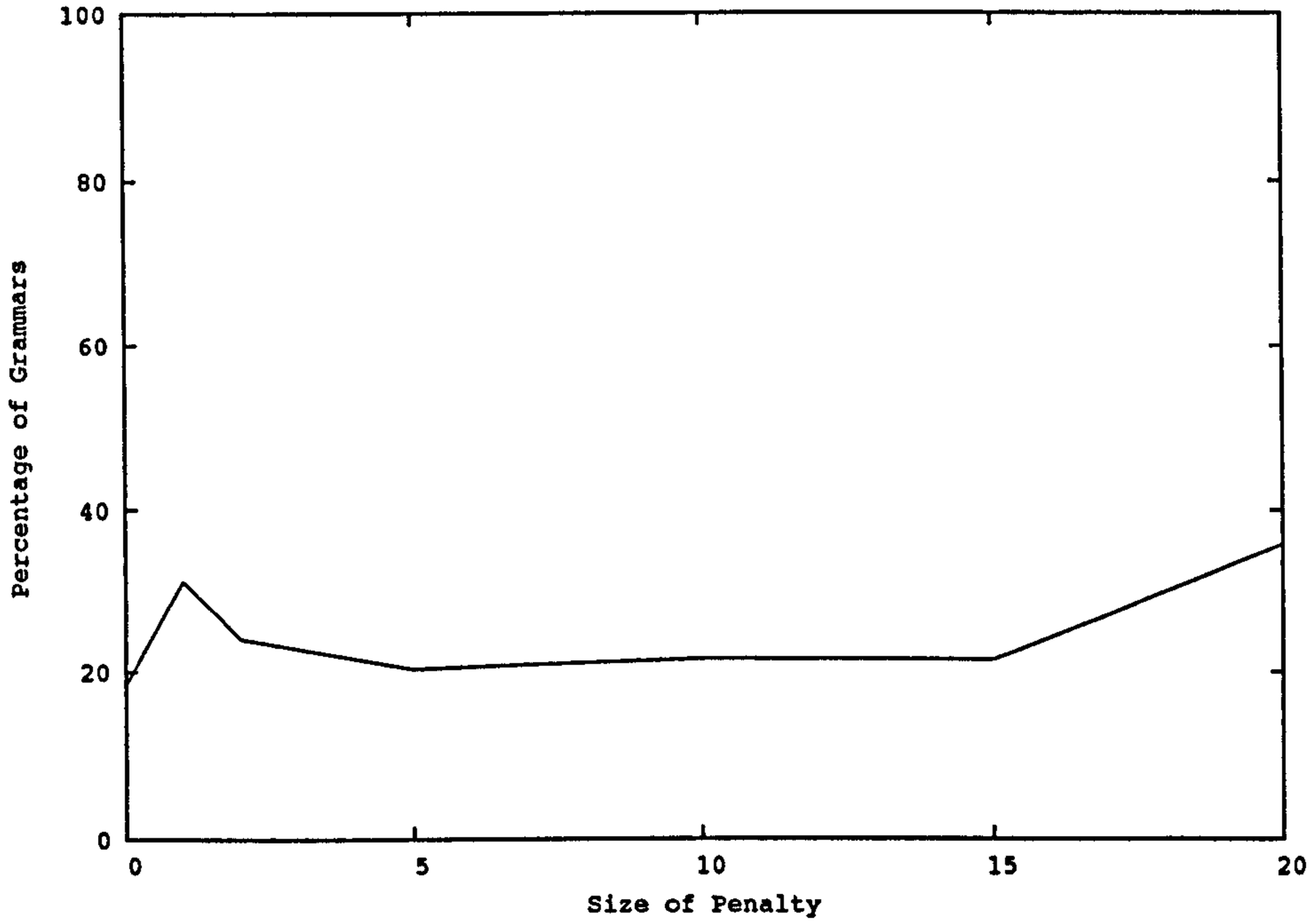

Figure 6.8: *The percentage of simulations resulting in grammars containing a mixture of Type A and Type B rules for a variety of different penalty values.*

in number, being recruited mostly from the pool of simulations resulting in "mixed" grammars. However, this is only effective to a point, and once the penalty gets too large, the mixed grammars start to increase in number again, at the expense of those containing only Type B rules.

Thus, when an agent is intolerant of ambiguous utterances, the introduction of a penalty does appear to create selective pressure for grammars containing only Type B rules, as shown in the table below:

penalty	Type A (all top level rules contain 1 noun category)	Type B (all top level rules contain 2 noun categories)	mixed (a mixture of 1 category and 2 category rules)
none	43.66	38.03	18.31
1	43.24	25.68	31.08
2	43.66	32.40	23.94
5	34.78	44.93	20.29
10	35.38	43.08	21.54
15	41.43	37.14	21.43
20	41.79	22.39	35.82

However, the proportion of two-noun category grammars is still less than 50%, and still not any higher than that achieved when the simulation was first implemented using a probabilistic grammar and 1% chance of re-ordering, with ambiguity simply ignored rather than rejected (where we achieved 43.14% Type B grammars). Furthermore, the introduction of intolerance to ambiguity results in a large decrease in the *regularity* of grammars causing very few to converge on a single set of terminal and non-terminal categories in their top-level rule.

6.2.4 Highly irregular grammars

It is possible to observe the decreased regularity by other measures too: when ambiguity is tolerated, in simulations using the probabilistic parser, and re-ordering utterances with a probability of 1%, a significant number of the emergent grammars show optimal behaviour: that is, a grammar that contains only a single fully lexcialised top level rule and its permutations, plus one lexical entry for each of the *individuals* and *actions* in the meaning space (or in the case of Type B grammars, two entries for each individual,

Figure 6.9: *The percentage of completed runs of the simulation which result in grammars with optimal behaviour, either of Type A or Type B, over a range of penalty values.*

one for the noun category used to express the subject of the sentence, and another for the noun category used to express the object). A total of 46.87% of the simulations runs exhibit this optimal behaviour, and of those optimal grammars, just under a third are of Type B. So already there is a disparity in the degree of optimality displayed by each grammar type. Although Type A grammars are only slightly more numerous than Type B under these conditions, *optimal* Type A grammars are *much* more common.

When an intolerance of ambiguity is added to the equation, optimal behaviour suddenly becomes an infrequent occurrence. Just this simple change causes it to drop dramatically from 46.87% of all simulations, to only 8.89%. Introducing a penalty for producing a string that can be misinterpreted results in a slight recovery in the number of optimal grammars emerging, but

not to its former level, and only for lower penalty levels. By the time the penalty reaches a size of 5, the percentage has started to decline again, as shown in Figure 6.9. What is particularly noteworthy, though, when looking at the percentage of converged runs that result in optimal grammars, is that *all but one* of these grammars are of Type A. Once intolerance of ambiguity is introduced, an optimal Type B grammar becomes exceedingly rare indeed.

Another measure of a decrease in the regularity of the grammars that emerge from the system when agents are intolerant of ambiguity (in the case of Type B grammars) is whether or not each of the two noun categories is used consistently for one thematic role. That is to say, given two noun categories, 1 and 2, if category 1 is used to express the subject of the sentence in one rule, is it also used to express the subject in all other top-level rules, or do some of them use it as the object?

When ambiguity is tolerated, and agents use a probabilistic parsing algorithm, in conjunction with re-ordering of utterances with a probability of 1%, non-terminals representing nouns are used consistently in 75% of those grammars which can be classified as having converged on Type B. When intolerance is introduced, this drops to 18.75% of converged Type B grammars. This behaviour is not reversed by the introduction of a penalty for producing strings with multiple interpretations.

Thus it would appear that the use of penalties *does* create a selective pressure for the emergence of Type B behaviour, but that this is at the expense of regularity.

6.3 Applying Penalties Universally to all Permutations of a Rule

The changes that have been made to the model, that is, the introduction of a penalty when utterances are produced that are parsed by the hearer to give an incorrect meaning, are designed to remove from the grammar those rules which can result in ambiguity, or at least significantly reduce the probability of their use. Primarily, it is expected that penalties will mostly be applied to those rules containing a single noun category used to represent both the subject and object of a sentence. Whenever the re-order procedure is invoked on an utterance resulting from such a rule, and happens to cause the inversion of subject and object, ambiguity will result, and the rule will be penalised. However, this neglects the fact that this rule may have other permutations in the grammar, in which subject and object have not been inverted, and which thus do not result in ambiguity. In an attempt to address this, the penalty mechanism was altered so that it is not just applied to the rule which *created* the ambiguous utterance, but also any other rules in the grammar which contain the same non-terminal and terminal characters in a different permutation. By penalising all permutations of a rule in this way, it is hoped that we will be more successful in removing single-noun category rules from the grammar.

Figure 6.10 shows a range of penalty values, and for each, the percentage of simulations which result in grammars that have converged on a grammar that can be classified as either Type A or Type B according to our original definition. These results appear to show that when penalties are applied to all permutations of a rule, the trend is similar to when they are applied only to the instance which caused an ambiguous utterance to be formed: that is, a decrease in the number of simulations converging on grammars of one type or the other either type as the penalty size increases.

Figure 6.10: *The percentage of simulations resulting in converged grammars for a range of penalty values when the penalty is applied to all permutations of a given rule, not just the one that resulted in the ambiguous utterance that has been observed.*

Looking at the distribution of Type A and Type B grammars amongst those runs which *do* converge would suggest that such an application does not seem to further our aim of promoting separate noun categories to express subject and object of the sentence, as demonstrated in Figure 6.11.

This is not promising at all: for all penalties, Type A grammars seem to far outnumber those of Type B, actually *gaining* in predominance as the penalty size increases, for low values. In the range of penalties that were producing the best results when only a single rule was being penalised (i.e. between 5 and 10), the number of Type A grammars is nearly 80%.

However, it may be recalled that, looking only at those grammars which can

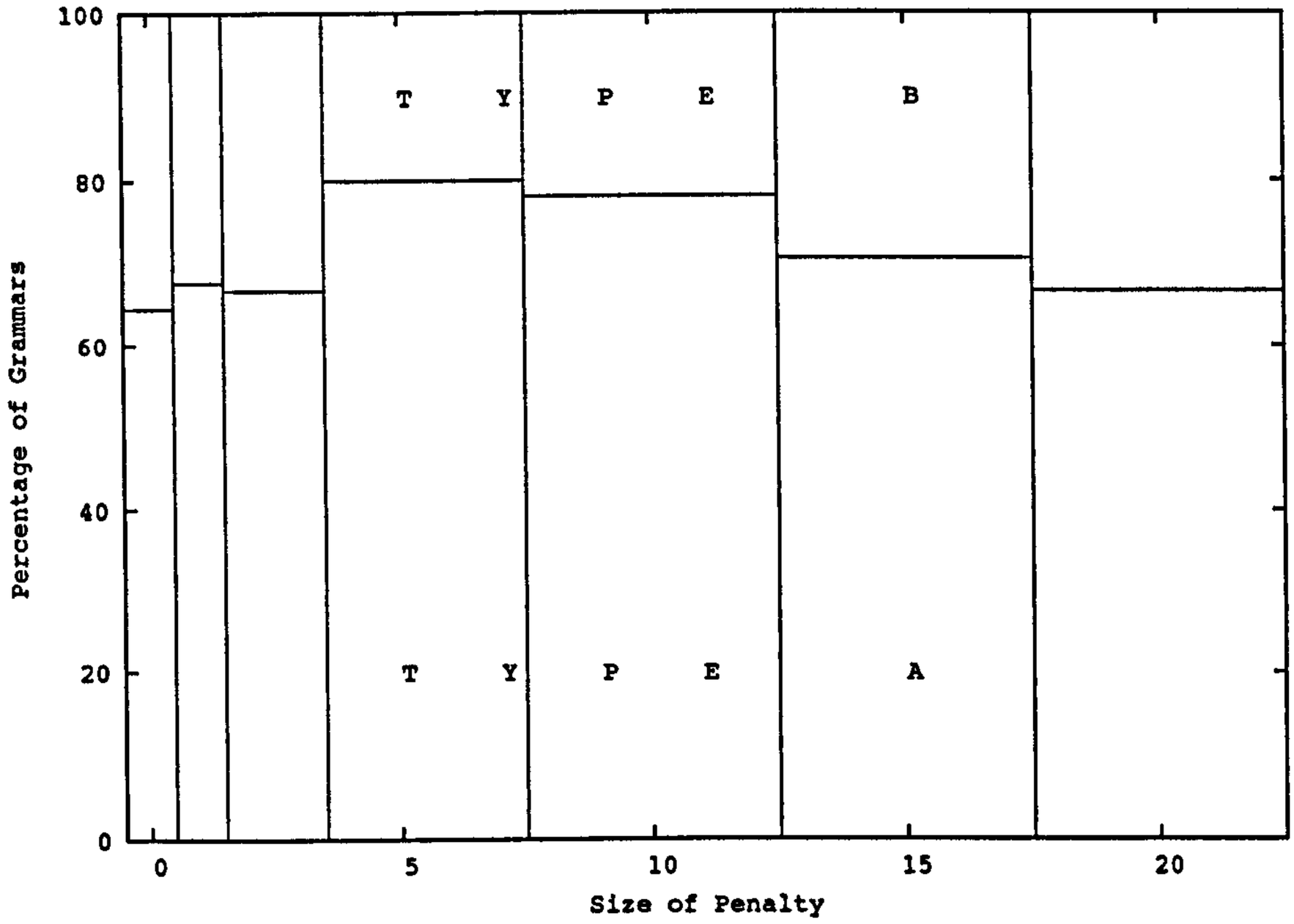

Figure 6.11: *The relative proportions of simulations resulting in Type A and Type B grammars as a percentage of those simulations which do converge on a grammar, when penalties are applied to all permutations of a rule, rather than just one form of it.*

be considered to be of either Type A or Type B according to our original definition was in some ways deceptive, as it disguised a trend towards grammars which only use Type B rules, but are somewhat irregular, and thus do not necessarily use the same three non-terminal categories in each rule. Is the same thing happening again?

Figures 6.12 and 6.13 show the number of simulation runs resulting in grammars containing only Type A rules, and only Type B rules respectively, as a percentage of the total number of completed simulations.

Here we can see that the number of simulations resulting in grammars containing only Type B rules is consistently lower than those containing only

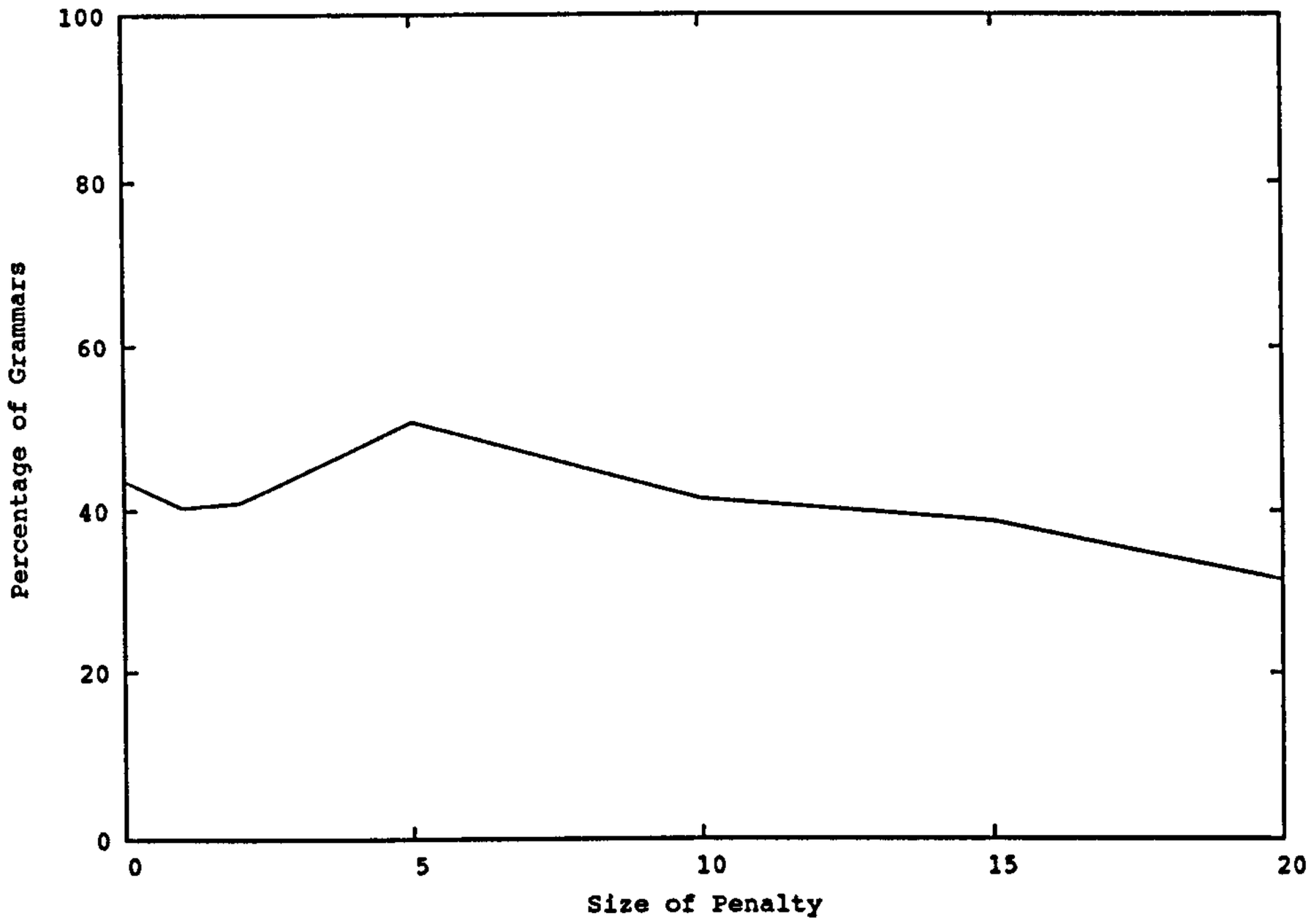

Figure 6.12: *The number of simulations resulting in grammars which only use Type A rules, as a percentage of the total number of completed simulations, over a range of different penalty values. Penalties have been applied to all permutations of a given rule, not just that being used when the ambiguity being rejected occurred.*

rules of Type A. Furthermore, there is something of a decrease in the former when a penalty is introduced, which is mirrored by an *increase* in the latter for low penalty values, peaking at a penalty value of around 5. This is exactly the value that seemed to be an optimal penalty when being applied *only* to the rule which was directly implicated in the production of an ambiguous utterance. And yet when applied to *all* permutations of that rule, it seems to have the opposite effect: use of two separate noun categories, one to represent the subject of the sentence and the other to represent the object becomes increasingly unlikely. However, the number of simulations resulting in mixed grammars continues to increase with penalty size, this time at the expense of Type A grammars. Thus, contrary to expectations, the application of the

Figure 6.13: *The number of simulations resulting in grammars which only use Type B rules, as a percentage of the total number of completed simulations, over a range of different penalty values. Penalties have been applied to all permutations of a given rule, not just that being used when the ambiguity being rejected occurred.*

penalty across the board, to all rules that are *permutations* of the one which was the source of ambiguity, seems apparently to lessen the pressure towards two-noun category grammars that we are seeking to emulate.

6.4 Modifying the bottleneck

As in Section 5.4 further experiments were carried out in order to investigate the effect of manipulating the bottleneck of transmission on the results of these simulations. As before, bottlenecks were manipulated by selecting at

random a subset of the entire meaning space at the start of each generation, and allowing speakers to produce utterances only for the meanings within this subset. A range of different subset sizes was used, from 30% to 90% of the meaning space. In addition to this change, the number of utterances made per generation was increased from 100 to 1000 in order to give learners a large pool of utterances to sample from, in the hope that this would help them to fully exploit the statistical effect of the quasi-probabilistic parser.

Recall that in Section 6.4 it was found that externally imposing a bottleneck in the absence of any word order freedom had a very favourable effect on the degree of compositionality emerging from the simulations. The greatest effect was seen for bottlenecks of between 20% and 30% of the meaning space, where the number of simulations failing to converge on a grammar was dramatically reduced and the proportion of the grammars emerging from successful runs that showed optimal characteristics also increased by a large amount.[2]

What was perhaps a little surprising was the effect this had on the distribution of Type A and B grammars. For very tight bottlenecks, where only a small subset of the meaning space was in use during any given generation, there is a massive swing towards Type A grammars. This is probably to be expected because a Type A grammar is the optimal form of a fully compositional language. However, what was less predictable was the fact that at slightly more relaxed bottlenecks, Type B grammars were favoured. This was especially true for the range of subset sizes between 60% and 90% of the meaning space, which is particularly interesting because this is the size of bottleneck that the original simulation, with its coverage of 0.6339 falls into,

[2]To recap: optimal grammars were defined as containing only a single top-level rule for creating sentences, plus one rule for each item in each syntactic categories: for Type A grammars (containing only a single noun category used as both subject and object of the sentence) this gives a total of 11 rules, and for Type B grammars, (which have two distinct noun categories, one of which is used to represent the subject of the sentence and the other the object) a total of 16 rules.

and yet the results are very different.

Once freedom of word order was introduced, in the form of chance re-ordering events at a 1% probability, things changed greatly. Unfortunately, due to practical constraints with runs failing to complete within a realistic timescale, it was not possible to test the full range of bottleneck sizes, but for values where between 20% and 60% of the meaning space was in use, there does seem to be a swing towards the emergence of Type B grammars as the bottleneck is relaxed.

What will happen if we try this approach in conjunction with the work on rejecting ambiguity described in this chapter? A further set of simulations were run in with agents using the quasi-probabilistic parser and making 1000 utterances per generation as in Section 5.4. Once again, the probability of re-ordering was set at 1%. This time, however, whenever the speaker produced an utterance that was ambiguous to the learner, the rules were penalised as described in Section 6.2.2. The penalty value was set to 5. Bottlenecks allowing agents to use from 20% to 60% of the meaning space were applied.

As with previous runs, increasing the number of utterances per generation to 1000 in conjunction with the use of re-ordering resulted in large numbers of simulations failing to run for the full 5000 generations. The table below gives the percentage of simulations that had to be halted for each subset size:

subset size	percentage of simulations terminated
20%	0.00
30%	50.00
40%	57.69
50%	73.08
60%	65.38

Of those simulations which did complete, the percentages resulting in Type A grammars, Type B grammars and those not converging are given below:

subset size	Type A (1 noun cat)	Type B (2 noun cats)	non-converging
20%	0.00	0.00	100.00
30%	69.23	15.38	15.38
40%	81.81	9.09	9.09
50%	14.29	57.14	28.57
60%	0.00	88.89	11.11

For a subset size of 20% of the meaning space, all the simuations run failed to converge on a grammar of either type. Rule systems remained almost entirely holistic and very little compositionality was seen. For subset sizes of 30% upwards, compositional behaviour was seen, and simulations resulting in both Type A and Type B grammars did occur. Once again, due to the very small numbers of simulations actually completing, it is difficult to draw any firm conclusions from these results, but it does appear that tight bottlenecks select for Type A grammars, and when middling-sized bottlenecks are employed, Type B grammars are favoured. When a subset size of 60% was used, all but one of the simulations completing were of Type B grammar. The one that was not of Type B did not converge on a grammar of either type.

6.5 The overall picture

So, is it possible to draw any conclusions from the somewhat confusing picture presented by these results?

One thing that is abundantly clear, is that causing agents to reject utterances that can be misinterpreted, instead of simply accepting them and assuming

that the perceived meaning was the intended one, causes a great deal of grammatical instability. Fewer simulations run to completion, fewer of those which do complete converge on grammars of *either* Type A or Type B, and fewer of the grammars which do emerge are optimal or use their two noun categories (if they have two) in a consistent manner (that is, with one category *always* expressing the subject of the sentence and the other *always* expressing the object).

When a small penalty is introduced which reduces the counts on the rules used to create the misinterpreted utterance, these problems are exacerbated further. However, increasing the penalty size does seem to go some way towards redressing the balance. This is effective up to moderate penalty values of between 5 and 10. Beyond this, the destabilising effect returns.

However, the use of a penalty does indeed appear to create some degree of selective pressure for grammars using two distinct noun categories, one to express the subject of the sentence and the other to express the object. This is masked somewhat by the disorder that rejecting ambiguity causes in the first place – because there are fewer simulations resulting in what we consider to be of either Type A or Type B according to the original definition used, the trend only becomes apparent when we look inside the grammars, at the *types* of rule each contains. Then we can see that for low to medium sized penalty values, as the size of those penalties increases, the number of grammars using only rules that contain two noun categories also climbs. At high penalty values, the number of such grammars declines again. Once more, the optimum penalty value appears to be somewhere between 5 and 10.

However, this does not really represent much of an improvement over the earlier version of the model in which ambiguous utterances were simply ignored: although the proportion of grammars with two noun categories has increased relative to those with only one, the absolute number is actually very simi-

lar – 44.93% of all simulations with ambiguity rejected and a penalty of 5, compared to 43.14% when ambiguity is ignored. Furthermore, the *quality* of the grammars emerging is much poorer: they are *messier* and much less optimal. Attempts to *tidy up* the grammars by applying penalties to all permutations of a rule, (as opposed to just the version that resulted in the offending ambiguity) were not successful.

The most significant improvement was achieved by using a bottleneck of 60% of the meaning space, in conjunction with a penalty of size 5, and agents making 1000 utterances per generation, resulting in 88.89% of the grammars emerging being of Type B. This seems to reinforce the earlier impression from Chapter 5 that very tight bottlenecks favour Type A grammars, whilst slightly more relaxed ones favour Type B. This is perhaps not an unlikley scenario, given that Type B grammars are somewhat *suboptimal* in terms of compositionality: whilst they are fully compositional in the sense that the meaning of an utterance is a function of the meaning of its parts and the way they are combined, the very fact that they have two separate categories to represent nouns means that agents will have to observe a larger portion of the meaning space in order to be able to acquire the grammar in its entirety. Thus under circumstances where the bottleneck is extremely tight, it is quite likely that the pressure for a concise and minimal language that can be learnt in as few observations as possible will outweigh any pressure for distinguishable subject and object categories that we have succeeded in creating with our manipulations such as introducing word order freedom or penalising speakers for producing ambiguous utterances.

6.6 Discussion

The prime aim of this chapter was to build on attempts made in the previous one to use freedom of word order to promote the emergence of a primitive

form of case system. This was achieved by specifying that learners should reject any utterance which can be parsed by the rules in their own grammar to give a meaning other than that intended by the speaker. Speakers are then required to apply a penalty to the counts associated with the rules that were used to create the utterance, and to find an alternative way of expressing the required meaning.

In conclusion, it seems necessary to say that these changes have not resulted in a significant swing towards case-like behaviour.

As discussed in Section 6.1.2, the original intervention made in Chapter 5, of introducing word order freedom by calling a *re-order* procedure on the speaker's utterances with a fixed probability, initially 1%, *appears* to create a selective pressure for the emergence of the case-like behaviour we are seeking. However, in the absence of any requirement for agents to be able to understand each other, and thus any need to be able to resolve ambiguities caused by variations in word order, it seems likely that this effect is artefactual, and what is actually happening is that the introduction of the "disruption" caused by re-ordering results in a greater tendency of simulations to converge on the "suboptimal" Type B grammars.

Changes that require agents to make themselves understood (that is, causing rejection of ambiguous utterances, and applying penalties to the rules that created them) *do* appear to result in some pressure towards Type B behaviour, but only against a backdrop of much poorer performance. The very best result obtained with the penalty system is for 44.93% of grammars to contain only Type B rules, (this was achieved using a penalty value of 5), compared to 43.14% of simulations resulting in a full Type B grammars when ambiguity is tolerated and no penalty is applied. Thus, clearly, no significant gain has been made, and instead there has been a great cost in terms of a huge increase in the number of irregular, sub-optimal and unconverged grammars.

However, manipulating the bottleneck would appear to have some potential to overcome this. Although some of the problems that have blighted attempts to force agents to make themselves understood are also exacerbated by manipulating the bottleneck in this way, namely the fact that large numbers of simulations fail to run to completion, amongst those simulations that *do* complete there seem to have been some very promising results.

6.6.1 Can further improvements be made?

The major problem with the current work seems to be that, whilst the interventions introduced are effective to some degree in creating a selective pressure for distinguishable subject and object categories, they are also highly disruptive, and seem to interfere largely with the emergence of compositional grammars themselves. Clearly there are two approaches to resolving this problem: the first is to try and increase the selective pressure further still in order to overcome this disruptive influence; the second would be to try and remove the disruption. How can either of these ends be achieved? What areas can be identified as sources of weakness in the current model? Are there implementational choices that could have been made differently that might have encouraged the emergence of the case-like behaviour we have failed to achieve?

One major problem is the fact the random nature of the ambiguity being introduced by our "chance re-ordering events". In Section 6.2.1 it was discussed how rejecting utterances that the hearer misunderstands will cause the speaker to scan its grammar for another way of saying the same thing, and how this might result in the selection of a rule using different noun categories to express subject and object, a Type B rule, resulting in selective pressure for such grammars. However, it is entirely possible that the agent will select another Type A rule. If it does, there are two possible outcomes,

depending on the source of the original ambiguity:

- The ambiguity in the original utterance resulted from the invocation of the re-order procedure. In which case, the utterance created by the newly selected Type A rule is unlikely to result in any ambiguity, as another re-ordering event would have to occur for this to happen. With the probability of re-ordering at only 1 in 100, consecutive re-ordering events like this will be very rare. Furthermore, even if a re-ordering event *does* occur, there is no guarantee that an ambiguous utterance will be the outcome: as discussed previously, 50% of the potential word orders are perfectly distinguishable from each other, simply by the position of the verb and any terminal categories that the string contains. Of the other 50%, only word orders which are identical to those already allowed by an agent's grammar, except with subject and object inverted, will result in any ambiguity. If an agent only has a single word order rule in its grammar, then only 1 in 6 re-ordering events will cause the new re-ordering event to be rejected, and this proportion will be even lower if the single rule contains top level terminal categories (thus increasing the number of possible permutations).

- The ambiguity in the original utterance was *not* the result of a re-ordering event. If this is the case, it must be because the agent has already learnt an alternative word order with subject and object inverted, presumably from a previous re-ordering event, and succeeded in lexicalising the noun categories in that rule, *before* learning any rules for the actual word order exhibited by the speaker. When this happens, it effectively becomes impossible for the agent to learn the correct word order at all, and all utterances exhibiting it will be rejected. However, as the lexicalisation of non-terminal noun categories will have to have happened *before* the correct word order is observed, this scenario is actually incredibly unlikely, because it would require several re-ordering events, each resulting in inversion of the subject

and object of the sentence, before any utterances that have not been re-ordered are observed.

Thus, in the overwhelming majority of cases, once an utterance has been rejected, the replacement string will be accepted by the learner, regardless of whether it was created by a Type A or a Type B rule. The rule which produced the original ambiguous utterance will be penalised, and this in itself will result in some selective pressure for the use of Type B rules, but there will be nothing to stop any other Type A rule in the grammar from being propagated, so that pressure will not be as strong as it could be.

There are two potential approaches we could take to overcome this. The first would be to "carry over" the re-order event. If the string that is causing the ambiguity was created by a re-order event, then any alternative string that the agent produces will be re-ordered too. However, it is still highly likely that any utterance produced by Type A rules would be accepted under these circumstances because, as already discussed, at least 50% of the possible orderings that a re-order event can create will not produce any conflict – the strings produced by them will be perfectly distinguishable from the "opposite" meaning by the position of the verb. The only strings that will be rejected will be those which result in inversion of subject and object in relation to the word order used by any other rule already in the grammar.

The alternative approach would be to somehow *preserve* the word order choice made by the previous re-ordering event. This would require the "re-order" procedure to be somewhat more complex than it currently is: at the moment, it simply takes whatever elements are found in the top-level rule for composing a sentence, and re-orders them. It does not have any knowledge about or interest in whether those elements are terminals or non-terminals, and whether the rule is fully lexicalised or not, it simply treats them all alike. However, to be able to create the same ordering of subject, object and verb components when applied a different rule, which may or may not be

lexicalised to the same degree, and may or may not have non-terminals at the top level, extra information would be required. This could perhaps be achieved by creating agents which, rather than re-ordering, choose to *topicalise* one element of the meaning they are asked to express, by placing the lexicalised string associated with it (if one exists) in a specific position in the sentence. Thus, the same action could easily be repeated with the new string created by whichever alternative rules are chosen. However, this would potentially reduce the likelihood of subject-object inversion occurring, as it would limit the number of word order permutations that could occur. For example, if agent has a grammar whose basic word order is SVO, and topicalisation is achieved by bringing string representing the semantic element in question to the first position of the sentence, the only other possible orders are VSO and OSV, neither of which would result in ambiguity. However, if original word order were SOV, then topicalisation of the object would indeed have this effect, by resulting in an OSV sentence.

Another possible implementational weakness in our model as it currently stands lies in the way in which penalties are being applied. When an utterance is rejected, all the rules involved in the generation of the string associated with that utterance are penalised, including those lower down the parse tree, such as those which produce the substrings covered by the non-terminal categories. These rules are not themselves implicated in the generation of the ambiguity observed, and only represent different ways of expressing a given object or action within the meaning space.

How this might affect the results of our simulations seems a little unclear. In more optimal grammars, there is generally only one way of producing a "word" for a semantic element of a given category. Thus penalising the rule associated with this word will have little effect: when a choice needs to be made, it is still the rule that will be chosen, because no matter how low or high the count associated with it, it will still represents 100% of the possibilities.

However, in less optimal grammars, and earlier on in simulations, redundancy might occur, in which there are two strings of a given category with the same meaning. Under these circumstances, if one of them is penalised as a result of having been used to produce an utterance that resulted in ambiguity, then this will give the other a competitive advantage, and increase the likelihood of its being chosen in the future. This is somewhat undesirable, as in the case of ambiguity due to uncertainty over which noun in a given sentence is the subject and which is the object, neither of the two strings would have the power to disambiguate. One can forsee a situation in which the two forms engage in a *power struggle* of sorts, neither of them able to gain the upper hand and become the dominant form: one might manage to for a while, and then by chance will be selected for an utterance which will turn out to be ambiguous, causing the rule associated with that string to be penalised, and thus giving the other form a higher chance of being chosen for future utterances. This string might then assume dominance for a while until it finds itself in a similar situation, and the whole cycle is repeated, with neither form ever able to *win* and become the only possible way of expressing the given semantic element. Whilst this might not have any direct effect on the number of noun categories being used in the top-level rule, it is presumably a source of instability in the grammar, and may in part be to blame for the cost in terms of irregularity and sub-optimality that seems to be associated with rejecting ambiguity.

Of course, the whole method of reinforcing and penalising rules, and the nature of the quasi-probabilistic parser itself are somewhat adhoc, created through a series of small modifications to Kirby's original algorithm and perhaps not the best design for the job. One way forward might be to try a more principled method of probabilistic learning, such as that employed by Vogt [85] in his investigation of the emergence of compositionality in grounded language agents.

One final approach might be to try and create a *different* selective pressure for

the emergence of case other than the need to resolve ambiguous utterances: It could be speculated that the primary function of case is not to disambiguate. Multiple word orders do not make case-like behaviour *necessary*, but rather, the existence of case-like behaviour *facilitates* the use of multiple word orders, whilst case itself may have evolved to fulfil some other linguistic function. As discussed in mentioned in Section 3.4, Lupyan and Christiansen [53] have demonstrated that certain word orders are more easily learnt by a sequential learning device (in their case, recurrent neural networks) than others; however, when presented with languages with case markings, languages with a much wider range of word orders are learnable. More specifically speaking, the only un-marked word orders that were perfectly learnt by their networks were SVO and VSO. Addition of case markings brings SOV languages (which is actually the most common word order amongst natural languages) within the sphere of those which are completely learnable. This is in-keeping with the Greenberg's universal number 41 [38] which states that the vast majority of SOV languages exhibit case, whilst the majority of those languages which turn out to be case-less have an SVO or VSO ordering.

Lupyan and Christiansen postulate that the reason for this is due to the difficulty in interpreting unmarked nouns prior to the verb: it is the verb that provides grammatical information about the number of nouns a sentence should contain (by whether it is transitive, di-transitive etc.), and what semantic roles are required (agent, theme etc.). However, in an SOV language, the verb is received last, and the nouns must be stored in working memory till the end of the sentence before roles can be assigned to them. With case-markings, the role is made explicit. Similarly, those word orders which they found to be easily learnt without case markings are those in which the verb features very early in the sentence, and thus the expected roles are already known before subsequent nouns are encountered.

Of course, the current model is not a sequential learning device, and thus considerations regarding working memory, and the order in which elements

of the sentence are presented to the learner have no bearing here. This is why, in the experiments described in this thesis, all word orders are equally likely to emerge, and equally easy to learn, regardless of whether they exhibit case-like behaviour or not.

Further results from Lupyan and Christiansen's study show that languages with free word order but a full case system, and languages with a strict word order and no case system are equally easily acquired: the learner has no need for a fixed word order once case markings are present. This adds weight to the idea that it is case markings make freedom of word order possible, rather than existing to disambiguate between potentially conflicting word orders as discussed above.

6.7 Summary

In the current chapter we have attempted to promote the emergence of case-like behaviour by applying penalties to those grammar rules which result in utterances where the hearer is unable to successfully decode the intended subject and object of the sentence, in the hope of increasing the pressure for distinguishable subject and object forms that we tried to introduce in Chapter 5 through the introduction of freedom of word order. Although we have managed to achieve a moderate degree of success, particularly when externally manipulating the bottleneck of language transmission, the results presented are far from dramatic. We have discussed the possible reasons for this, both in terms of our implementation and the role that case may play in the learning and comprehension of language. However, the case-like behaviour that we have tried to model here has been very primitive, in the form of distinct noun categories being used to express subject and object of a sentence. In the next chapter, we will change the focus of our investigations slightly. We will move away from the idea that word order freedom is a

driving force for the emergence of case, and instead concentrate on trying to achieve proper inflectional case markings within fixed word order languages, which might themselves *facilitate* the use of a wider range of word orders.

Chapter 7

Simulating Inflectional Endings

Chapters 5 and 6 described attempts to use word order freedom as a driving force for the emergence of case-like nouns. These attempts were not terribly successful: overall a modest increase in the number of simulations displaying the desired behaviour was seen in Chapter 5, but further analysis of the situation seemed to indicate that this was not actually due to the use of word order freedom, but in fact other changes to the model that were necessary in order to facilitate it. In Chapter 6, attempts were made to address this, but no significant further increase in the behaviour being sought was achieved. Furthermore, the type of "case" system which did emerge was not terribly convincing, in that "nouns" of one particular type would often be used to express the subject of the sentence according to one grammar rule, and the object according to another.

Another shortcoming (albeit an anticipated one) of the results presented so far, is the *nature* of the case-like behaviour that we were attempting to promote. Whilst English is notoriously poor morphologically, many other natural languages have much richer morphologies, including the use of inflectional case endings which determine the roles of nouns referred to in the

sentence, as in the following example:

Canis hominem mordet
Dog NOMINATIVE man ACCUSATIVE bites
'The dog bites the man'

Canem homo mordet
Dog ACCUSATIVE man NOMINATIVE bites
'The man bites the dog'

In the above Latin sentences, the notion of a dog is expressed by the word "canis" in the first, and "canem" in the second, whilst the man is expressed by "homo" first, and then by "hominem". In both cases, the two words share a common stem, and but have different inflectional affixes, and it is these affixes which determine the roles each is playing in the event being described. Thus, although the *word order* is the same in both sentences, the meaning of each is quite different. The fact that the dog is doing the biting in the first sentence and being bitten in the second is indicated by the inflectional ending *-is* for nominative case and the ending *-em* for accusative case respectively.

Clearly, to whatever extent that we can consider case to have emerged in the results of the previous chapters, this notion of stem plus affix is completely absent. The behaviour seen can at best be described as some form of primitive "proto-case": each individual in the meaning space is represented by two distinct strings, one used when it is the actor or originator of the event being described, and the other when it is the thing being acted upon. These two strings are completely unrelated, and in many cases are quite different.

It is proposed in the current chapter is to modify the simulations described so far in an attempt to achieve grammars exhibiting a proper inflectional system of case marking, using a common noun stem, to which an inflectional affix

is added to indicate the role of the individual. The intention is to abandon the idea of using freedom of word order as a selective pressure for case-like behaviour, on the basis that, in-keeping with Jackendoff's theory of the evolution of language [42] as well as the results of Lupyan and Christiansen [53] described briefly in Chapter 6, it is seems quite likely that case evolved *first*, and that the use of free word order was then facilitated by its existence.

In Section 7.1 I shall present a discussion of the possible reasons why inflection cannot emerge in the current model, and go on in Section 7.2 to suggest possible changes which might enable it. Finally, in Section 7.4, the results of the modified system are described, followed by a discussion of their significance in Section 7.6.

7.1 Limitations of the Current System

The desired case system where nouns are made up of a common stem plus inflectional affixes cannot emerge within the current system as the inducer used in the work described previously is not capable of *effectively* learning inflectional grammars. To illustrate this point, the reader is asked to imagine a "toy" language something like English, but incorporating the inflectional marker *a* to indicate which participant is carrying out the action described in a given sentence, and the marker *b* to indicate the individual that is being affected by the action. These can be seen as markers denoting the subject and object of the sentence. Thus a string from this language representing the the English statement "John loves Mary" would be *j,o,h,n,a,l,o,v,e,s,m,a,r,y,b*.

What happens if we present sentences drawn from this language to the current implementation of the inducer? As before, sentences are presented as string-meaning pairs, where the meaning is represented by a three-place vector, with the roles of participants implied by position in the vector: the first

position represents the action or event being described, the second the actor or originator of that event and the third the individual being acted upon.

As described previously, the inducer takes each utterance presented and creates a "holistic" production rule for it. Thus if we present the sequence of utterances covering the four meanings [loves, john, mary], [loves, bob, mary], [loves, john, bob] and [hates, john, mary], we can follow the induction process as the grammar is built.

The first two utterances presented are

j,o,h,n,a,l,o,v,e,s,m,a,r,y,b meaning [loves,john,mary]
and
b,o,b,a,l,o,v,e,s,m,a,r,y,b meaning [loves,bob,mary]

causing the following two rules to be added to the grammar:

1 s/[loves,john,mary] ⟶ j,o,h,n,a,l,o,v,e,s,m,a,r,y,b
2 s/[loves,bob,mary] ⟶ b,o,b,a,l,o,v,e,s,m,a,r,y,b

Rules are then compared on a pairwise basis: if the semantic vectors representing the meanings of the two utterances differ by the value at one location, and the two strings on the right-hand sides of the production rules differ by a single substring, then the difference in meaning is attributed to the difference in the strings. Thus in the case of the two rules above, the string *j,o,h,n* is interpreted to mean "john" and the string *b,o,b* is interpreted to mean "bob" resulting in the following new rules (which replace 1 and 2 above):

3 s/[loves,X,mary] ⟶ NT1/X,a,l,o,v,e,s,m,a,r,y,b
4 NT1/john ⟶ j,o,h,n
5 NT1/bob ⟶ b,o,b

If the next utterance presented is

j,o,h,n,a,l,o,v,e,s,b,o,b,b meaning [loves, john, bob]

this will, again, initially result in the formation of a "holistic" rule as follows:

6 s/[loves,john,bob] ⟶ j,o,h,n,a,l,o,v,e,s,b,o,b,b

This rule will then be compared on a pairwise basis with the others in the grammar to see if any of the learning heuristics may be applied. Firstly the *subrule* operation will identify the substring *j,o,h,n* as being identical to the right-hand side of rule 4, and will change the rule to

6 s/[loves,X,bob] ⟶ NT1/X,a,l,o,v,e,s,b,o,b,b

This rule will then be compared with rule 3, and a single difference found: the semantic value in the object position of the meaning vector, and the substrings *m,a,r,y* and *b,o,b* respectively, resulting in the new rules

7 s/[loves,X,Y] ⟶ NT1/X,a,l,o,v,e,s,NT2/Y,b
8 NT2/mary ⟶ m,a,r,y
9 NT2/bob ⟶ b,o,b

and the removal of rule 3. This will be followed by the merging of categories NT1 and NT2, as rules 5 and 9 are identical but for the category name. Thus rule 7 becomes

7 s/[loves,X,Y] ⟶ NT1/X,a,l,o,v,e,s,NT1/Y,b

Finally, if presented with the utterance

j,o,h,n,a,h,a,t,e,s,m,a,r,y,b meaning [hates, john, mary]

after removal of the substrings "j,o,h,n" and "m,a,r,y" by the *subrule* operator, we are left with the following:

10 s/[hates,X,Y] ⟶ NT1/X,a,h,a,t,e,s,NT1/Y,b

which on comparison with rule 7, has only one difference – the value in the *action* or *event* position of the semantic vector, and the substrings "h,a,t" and "l,o,v".

Thus we end up with the grammar

10 s/[P,X,Y] ⟶ NT1/X,a,NT3/P,e,s,NT1/Y,b
4 NT1/john ⟶ j,o,h,n
5 NT1/bob ⟶ b,o,b
8 NT1/mary ⟶ m,a,r,y
11 NT3/loves ⟶ l,o,v
12 NT3/hates ⟶ h,a,t

This grammar contains a single noun category (NT1) which represents the noun "stem" and which is used interchangeably for subject and object roles. But the markers *a* and *b* are stranded in the top level rule. They are not related in any way to the noun to which they belong (other than by juxtaposition) or the syntactic category they specify, thus rendering them effectively meaningless. It should be noted that this grammar will be able to successfully parse and generate sentences from the target language, but clearly it does not capture it adequately as the meaning of the inflectional affixes has been lost.

Thus modifications to the inducer are needed to enable it to learn inflectional affixes more effectively, if the emergence of case marking is to be achieved.

7.2 Modifying the Learning Inducer

7.2.1 Similarities or differences?

One step towards this might be to build an inducer that works on the basis of *similarities* between utterances rather than *differences*. Thus, given two utterances which have the the individual *John* in the subject position of their meaning vector, represented by the sequence *j, o, h, n* in the string associated with each of them, then the maximal similarity between the two strings will be *j, o, h, n, a* because the subject of the sentence is always followed by the inflectional marker *a* in this language. An inducer which partitions up the string on the basis of similarities rather than differences will therefore *include* the inflectional marking associated with a given noun, rather than leaving it behind.

This change to the induction algorithm requires a reworking of two of the four heuristics for comparing pairs of rules – *findchunks* and *findchunk*. To recap, the ***findchunks*** heuristic is as follows:

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_1/m_2 \longrightarrow \sigma_2$
  if m_1 and m_2 differ only by values v_1 and v_2 respectively
  and σ_1 and σ_2 differ only by substrings λ_1 and λ_2 respectively
    add new rules $\mathcal{N}_2/v_1 \longrightarrow \lambda_1$ and $\mathcal{N}_2/v_2 \longrightarrow \lambda_2$
    where $\mathcal{N}_2 \in \mathcal{N}$
  replace r_1, r_2 with new rule $\mathcal{N}_1/m_3 \longrightarrow \sigma_5$
    where $m_3 = m_1$ with v_1 replaced by variable $\mathcal{V}$
    and $\sigma_5 = \sigma_1$ with λ_1 replaced by $\mathcal{N}_2/\mathcal{V}$

The heuristic *findchunk* is very similar, however, it requires m_1 and m_2 to differ by value v and variable $\mathcal{V}$, and σ_1 and σ_2 to differ by substrings λ and $\mathcal{N}/\mathcal{V}$ where $\mathcal{N}$ is a non-terminal category. The substring λ is then re-written in a new rule as an instance of that non-terminal. (A formal description of this heuristic can be found in Chapter 5).

These heuristics are replaced with the new heuristic ***findsimchunks***:

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_1/m_2 \longrightarrow \sigma_2$
  if m_1 and m_2 contain the common value v_1
  and σ_1 and σ_2 contain the common substring λ_1
    add new rule $\mathcal{N}_2/v_1 \longrightarrow \lambda_1$
      where $\mathcal{N}_2 \in \mathcal{N}$
    replace r_1 with new rule $\mathcal{N}_1/m_3 \longrightarrow \sigma_3$
    replace r_2 with new rule $\mathcal{N}_1/m_4 \longrightarrow \sigma_4$
      where $m_3 = m_1$ with v_1 replaced by variable $\mathcal{V}$
      and $m_4 = m_2$ with v_1 replaced by variable $\mathcal{V}$
      and σ_3, $= \sigma_1$ with λ_1 replaced by $\mathcal{N}_2/\mathcal{V}$
      and σ_4, $= \sigma_2$ with λ_1 replaced by $\mathcal{N}_2/\mathcal{V}$

7.2.2 Evaluating the modified inducer

It is important to ensure that the new inducer is capable of learning a compositional grammar. This was tested by presenting sentences drawn from an optimal grammar. Sets of 100 sentences were presented over a total of 20 trials; it appears that although the inducer *is* capable of learning such a grammar, this is crucially dependant on the order of sentence presentation. Previously unobserved semantic concepts must be presented in conjunction

with others that are also previously unobserved, as those concepts which have already been seen are simplified to variables using the *subrule* heuristic. In 3 of these trials, the minimal grammar was learnt, but in the remainder, at least one "semantic value" was not correctly lexicalized.

For example:

1 s/[P, X, Y] ⟶ 2/X, 4/P, 2/Y
2 2/pete ⟶ q, r
3 2/kath ⟶ o, p
4 2/john ⟶ k, l
5 2/anna ⟶ s, t
6 2/mary ⟶ m, n
7 4/hates ⟶ c, d
8 4/adores ⟶ e, f
9 4/kisses ⟶ i, j
10 4/sees ⟶ g, h
11 s/[loves, X, Y] ⟶ 2/X, a, b, 2/Y

Here we can see that a lexical entry for the verb *loves* has not been learnt. Furthermore, there are no additional sentence presentations that will allow it to be formed, as all possible nouns in the meaning space have already been learnt. Thus there is no utterance that can be presented which will have only one similarity to rule 11: any sentence containing the predicate *loves* will have its subject and object removed by the "subrule" heuristic, making it identical to rule 11.

In order to overcome this, an additional heuristic was created, to produce these lexical entries by comparing rules such as 11 with top level sentence rules like rule 1:

Given a set of rules $\mathcal{R}$ representing the grammar of agent a, and a set of

non-terminal symbols $\mathcal{N}$:

for $r_1, r_2 \in \mathcal{R}$ where $r_1 = \mathcal{N}_1/m_1 \longrightarrow \sigma_1$ and $r_2 = \mathcal{N}_1/m_2 \longrightarrow \sigma_2$
if m_1 contains value v and variables $\mathcal{V}_1$ and $\mathcal{V}_2$ (in any order)
and m_2 contains variables $\mathcal{V}_3$, $\mathcal{V}_4$ and $\mathcal{V}_5$
find variable $\mathcal{V}_x$ in m_2 which occupies the position
corresponding to value v in m_1
if σ_1 and σ_2 differ by substrings λ and $\mathcal{N}_2/\mathcal{V}_x$
add new rule $\mathcal{N}_2/v \longrightarrow \lambda$
replace r_1 with new rule $\mathcal{N}_2/m_3 \longrightarrow \sigma_3$
where $m_3 = m_1$ with v replaced by variable $\mathcal{V}_y$
and σ_3, $= \sigma_1$ with λ replaced by $\mathcal{N}_2/\mathcal{V}_y$

With this additional heuristic, when the trials described above were repeated, the inducer was able to learn the correct minimal grammar in 20 out of 20 trials.[1]

7.3 Allowing Further Generalisations

Having created an inducer that is based on similarities rather than differences, it would be instructive here to return to our hypothetical language discussed above (i.e. the one based on English, but including the inflectional markers "a" for subject and "b" for object). After being presented with a sufficient number of sentences drawn from this language, the new inducer would learn a grammar that looks something like this:

[1]Occasional misconvergences were observed, but not in any of the trials recorded here.

s/[P,X,Y] ⟶ NT1/X,NT3/P,NT2/Y
NT1/john ⟶ j,o,h,n,a
NT1/bob ⟶ b,o,b,a
NT1/mary ⟶ p,e,t,e,a
NT2/john ⟶ j,o,h,n,b
NT2/bob ⟶ b,o,b,b
NT2/mary ⟶ m,a,r,y,b
NT3/loves ⟶ l,o,v,e,s
NT3/hates ⟶ h,a,t,e,s

Unlike the grammar that would have been learnt by the original, difference based inducer, the similarities based inducer hypothesises two separate noun categories, NT1 and NT2, one of which is used to express the subject of the sentence (NT1) and the other used to express the object (NT2). Each "individual" in the meaning space is expressed twice, once as each of these categories, for example:

NT1/john ⟶ j,o,h,n,a
NT2/john ⟶ j,o,h,n,b

These pairs of rules are unmergeable;[2] although they have identical left hand sides other than the category name, the right hand sides of the two rules differ. Both contain string *j,o,h,n* but one is suffixed with the subject marker *a*, and the other with object marker *b*.

However, looking more closely at these two rules, it is clear that something is missing. The string *j,o,h,n,a* and the string *j,o,h,n,b* refer to the same individual, certainly, but they do not mean *exactly* the same thing, as would appear

[2]Recall that one circumstance under which the merge operation may be invoked is if changing the category name of the non-terminal on the left hand side of the rule would make the two identical.

from the semantics given – one represents John when he is the perpetrator of the action described in the sentence, and the other represents him when he is at the receiving end. The fact that these strings have been assigned identical meanings might suggest that they can be used interchangeably, when in fact this is not the case.

Furthermore, it is clear that regularities *between* the strings *do* exist – as previously observed, both noun forms referring to the individual John contain the common root substring *j,o,h,n*. How can the inducer be altered so that this information is captured?

7.3.1 An enriched semantic representation

When looking at sentences (in English) such as *John loves Mary* and *Bob sees Mary* it is clear that there is something semantically similar about the involvement of John and Bob in these sentences – they both represent the *perpetrator* of the actions described – the person doing the *loving* or the *seeing*; the actor. Whereas Mary in both cases is the person being *affected* by that action: the acted-on party.[3] This is reflected in the semantic representations used thus far in the simulations described by the fact that *John* and *Bob* share a common position in the vector, i.e. [loves, *john*, mary], and [sees, *bob*, mary]

Returning to our hypothetical case marked language: the string associated with the meaning [loves, john, mary] would be *j,o,h,n,a,l,o,v,e,s,m,a,r,y,b*. When comparison of this string with another results in association of the

[3]As in Chapter 4, one could conceivably use the terms *subject* and *object* here. However, I have chosen to avoid these, due to their syntactic nature, as I wish to avoid the implication that agents have any syntactic knowledge whatsoever. Role categorisations in this instance are being made on the basis of semantics: that is who is the *perpertrator* of an action or event (actor) and who the event is *happening to* (acted-on).

substring *j,o,h,n,a* with the semantic element "john" it is originally clear that this refers to John in his role as the perpetrator of the action because of his position in the semantic vector associated with the utterance. However, because this information is implied by position but never made explicit, it is lost once removed from its original context.

Thus in going from the rule

s/[loves,john,mary] ⟶ j,o,h,n,a,l,o,v,e,s,m,a,r,y,b

to

s/[loves,X,mary] ⟶ NT1/X,l,o,v,e,s,m,a,r,y,b
NT1/john ⟶ j,o,h,n,a

the fact that the string *j,o,h,n,a* represented the perpetrator of the action has been lost. The rules of the grammar are such that this string can only be "plugged back in" to this role, but the rule

NT1/john ⟶ j,o,h,n,a

itself does not contain this information: it is held in the distribution of the category NT1.

This results in a problem for our inducer when faced with pairs of rules such as

NT1/john ⟶ j,o,h,n,a
NT1/mary ⟶ m,a,r,y,a

or

NT1/john ⟶ j,o,h,n,a
NT2/john ⟶ j,o,h,n,p

In the first example, the two rules *appear* to have completely different semantics associated with them, despite the fact that both substrings, in their original context were associated with the *perpetrator* or *actor* in the event described, as represented in both cases by the suffix *a*.

In the second example, the two rules *appear* to have identical semantics associated with them, despite the fact that one substring, in its original context referred to John as the *actor* in the situation described and the other referred to him as the *acted-on*. However, both strings refer to the same entity, "John" as indicated by the common stem *j,o,h,n*.

Thus the creation of subrules, and removal of semantic elements from their initial contexts, renders generalisations between such pairs of rules impossible. For this reason, the semantic representation of sentences was augmented to make explicit the information that is held by position in the vector. Rather than a single vector, the representation is modified to be of a nested structure, showing the "role" of each participant as well as its value – the individual or action to which it refers. Thus the vector [loves, john, mary] becomes [[act,loves], [actor,john], [actedon,mary]].[4] And the vectors associated with the strings *j,o,h,n,a* and *j,o,h,n,b* would be [actor,john] and [actedon,john] respectively. This enables generalisations of the type that were not possible before, as the two strings now share the common semantic element "john" and the common substring *j,o,h,n*; a rule can now be formed relating the two. This rule is much more veridical than the previous two, as the string

[4]It might have been possible here to use the terms "agent" and "patient"; however I have chosen to avoid these as some of the "actions" used in the meaning space, such as *loves* and *hates* are non-agentive. However, despite the fact that the subjects of these sentences would generally be categorised as *experiencers* rather than agents, there does still appear to be something that they have in common with the agents of other predicates used, in the sense that they are still the *doer*. This was reflected in the original semantic representation by a common position in the semantic vector, and is similarly reflected in the new representation by the common designation, "actor". This is a point we will return to in Section 7.6.2.

j,o,h,n really does mean simply "john", and can be used in any context (*actor* or *actedon*). Similarly, the vectors associated with *j,o,h,n,a* and *m,a,r,y,a* would become [actor,john] and [actor,mary], again allowing generalisations that were not possible before: between the common semantic element "actor" and the common substring *a*.

The important thing to note about this augmentation of the semantic representation is that no information is actually being *added*; we are simply making explicit that which was previously implicit. We have noted that there appears to be some kind of semantic similarity between the roles of John and Bob in sentences such as *John loves Mary* and *Bob sees Mary* and chosen to encode this in our semantic representation; however, it was previously *implied* by the fact that in the semantic vectors representing these two events, John and Bob occupy the same "slot".

In order to support these changes to the semantic representation, the following changes to the inducer must be made:

- When computing the similarities between two semantic lists, the original *findsimchunks* heuristic requires any commonalities found to be atomic values rather than variables. However, with the new augmented semantic representation, it must allow non-atomic values, namely lists. Thus the heuristic was amended to check that the value *contains* no variables, whilst not necessarily being atomic itself.

- The original chunk-forming heuristics (*findchunks* and *findchunk*) operated only on pairs of rules with the same non-terminal on their left-hand sides. The same requirement was extended to the new *findsimchunks* heuristic when it was first introduced. In fact, for reasons of efficiency, all three heuristics were implemented so that they would only consider rules of category *s*, that is top-level rules. This was quite adequate with the original semantic representation, as the meaning space does

not include any nested concepts, so there was no possibility of needing to be able to induce grammars with a depth greater than one, making comparisons of other non-terminal categories unnecessary. However, the new semantic representation, by effectively adding another level of recursion, makes deeper structures possible. Furthermore, in order to make generalisations *across* roles as well as between them, the heuristic must be able to compare rules with differing non-terminals on their left-hand sides, such as

NT1/[actor,john] ⟶ j,o,h,n,a
NT2/[actedon,john] ⟶ j,o,h,n,b

By allowing allowing the heuristic to be applied to pairs of rules such as these, the common string *j,o,h,n* can be attributed to the the common part of the semantics, "john", and new rules created as follows:

NT1/[actor,X] ⟶ NT3/X,a
NT2/[actedon,X] ⟶ NT3/X,b
NT3/john ⟶ j,o,h,n

The *findsimchunks* heuristic has been amended accordingly: the requirement for the non-terminals on the left hand sides of rules to be identical was relaxed, and it will now compare any pair of rules.

- Due to the new possibility of deeper structures, changes to the *subrule* heuristic have also been necessary. It is now possible for rules that are not "top-level" to also contain non-terminal categories on their right-hand sides. Thus it is not possible to simply look at the potential "subrule" and determine whether the string on its right-hand side is a substring of the right-hand side of the superrule. Instead, a deterministic parser[5] has been added, to find all possible expansions of the rule and determine whether any of those are substrings of the "superrule".

[5]It would have been possible to use the probabilistic parser already built into the system here, but it was deemed somewhat inefficient, since the task in hand is merely to find all possible parses and determine whether *any* of them are appropriate.

For example, when comparing the rules

1 s/[[act,loves],[actor,john],[actedon,mary]] ⟶ a,b,c,d,e,f
2 NT1/[A,B] ⟶ NT2/A, NT3/B

if rules exist such as

3 NT2/john ⟶ c
4 NT3/actor ⟶ d

then NT/1 can be expanded to the string *c,d* meaning *[actor,john]* and it can be seen that rule 2 is indeed a "subrule" of rule 1 according to the requirements of the *subrule* heuristic.

Having made these changes to the inducer to accommodate the new semantic representation, the next important question is, can it still learn an optimal compositional language? The trials described above were repeated once again, and the inducer was able to learn exactly the same grammar for each sequence of sentence presentations as it had earlier. Thus ability to learn an optimal language has been in no way deteriorated by these further changes.

In addition to this, a series of trials were performed in which the inducer was presented with sequences of utterances from languages with varying degrees of inflection, using the augmented semantic representation.

The first language contained inflectional markings on only one of the noun roles, in this case the object: here the inducer was able to learn a grammar to cover this language, including attributing the correct meanings to noun-stem and inflectional affixes, in 19 out of 20 runs. In the run which did not fully converge on a minimal grammar, the inflection part had been learnt, but the noun stem had not, producing rules such as:

9/[A,mary] ⟶ m,n,6/A
9/[B,john] ⟶ k,l,6/B
9/[C,kath] ⟶ o,p,6/C
6/actedon ⟶ o,b

It is interesting to note that the number of rules in this apparently *suboptimal* grammar is fewer than that the number in the grammars evolving in the other 19 trials (17 rules versus 18).

The second inflectional language tested included inflectional markings on both subject *and* object roles: this complicates the issue slightly. If we present the inducer with sentences such as

k,l, s,u,b, a,b, m,n, o,b
which can be glossed as:
john actor loves mary acted-on

it will sometimes interpret the substring *s,u,b* as being a prefix associated with the string *a,b* meaning "loves", rather than a suffix attached to the string *k,l*.

For example, if the next observed utterance which shares one semantic value with the sentence above has "john" in the role of actor, then the largest common substring will be *k,l,s,u,b* and the inflectional marking will be associated with the noun as it should be. However, if the common value is the action, such as in the utterance *o,p,s,u,b,a,b,q,r,o,b* meaning [[act,loves], [actor,kath], [actedon,pete]], the largest common substring will be *s,u,b,a,b* which is interpreted as meaning [act,loves]. Sometimes such attributions are learnt alongside the correct ones. Thus, when presented with sequences of 100 sentences from this language, the inducer learnt a grammar in keeping with an "inflectional suffix on the agent" interpretation in 8 out of 20 trials, and learnt a complete grammar for this interpretation, but which included a

few rules from the "inflectional prefix on the verb" interpretation in a further 4 trials. In 4 of the remaining trials, it learnt a grammar for the "inflectional prefix on verb" interpretation, and the result of the final 4 trials was a mixture of rules in which neither grammar type were able to cover the entire meaning space.

It should be noted that the grammars induced during these trials for both "interpretations" of the affix *s,u,b*, (whether as a suffix of the noun, or part of the verb), are "correct" in that they cover all and only utterances from the language. Thus production and reception behaviour of agents with both grammar variants will be identical – it is just the internal structure of the language which differs. In particular, the meaning attributed to the various affixes.

Having satisfied ourselves that the changes to our model now allow it to effectively learn the types of languages we would like to see emerging, it is time to investigate the kinds of behaviour that are seen when it is employed in an iterated learning context.

7.4 Employing the new inducer in the Iterated Learning Model

Firstly the new inducer was tried with the original unaugmented semantic representation, to ensure that simple non-inflectional languages can indeed evolve using such an induction system.

Simulations of 5000 generations, with 100 utterances per generation were performed, as before, using the deterministic parser. It quickly became apparent that the problem of increasing rule length and multiple top level rules occasionally encountered when using the difference based inducer (and as

described in [71]) is much more prominent with the similarities based version. When attempting a run of 10 simulations, of 5000 generations, all but one of them failed to run to completion in a time of 15 hours. Inspection of grammars produced by these runs showed large numbers of top-level rules containing many (often repeated) non-terminal symbols. Thus to avoid this problem, a working memory limitation was added to the parser, whereby it effectively ignores any rules with a left hand side of more than 8 characters in length.

With this change in place, compositionality similar to that seen in the difference based inducer does indeed emerge: nominal and verbal categories are combined to form utterances whose meanings are made up of the meanings of their parts. As in Kirby's study on which this work is based [44], and earlier experimentation with the difference based inducer in Chapter 4, the grammars emerging from the simulation can be shown to make the transition from almost entirely holistic utterances in the first instance, to those with some compositionality as the agents have been able to make a small number of generalisations over chance similarities, to much more fully compositional systems, in which, in the optimal scenario, there will be a single rule governing the structure of a sentence. Figure 7.1 shows the number of meanings that agents can express at each generation (from a total meaning space of 100), as well as the size of the grammars. These results are averaged over 26 runs. Figure 7.2 shows each of the runs separately over the first 50 generations. It is clear that sufficient compositionality to be able to express almost 100% of the meaning space emerges very quickly: for the first generation, the number of meanings expressible is generally just above 50%, but this quickly climbs to in excess of 90% within the first five generations.

Returning to figure 7.1, we can see that the degree of compositionality continues to increase long after the point at which the entire meaning space can be expressed, as the average size of grammars continues to decrease, until it reaches a steady value of around 20. This slightly high value indicates

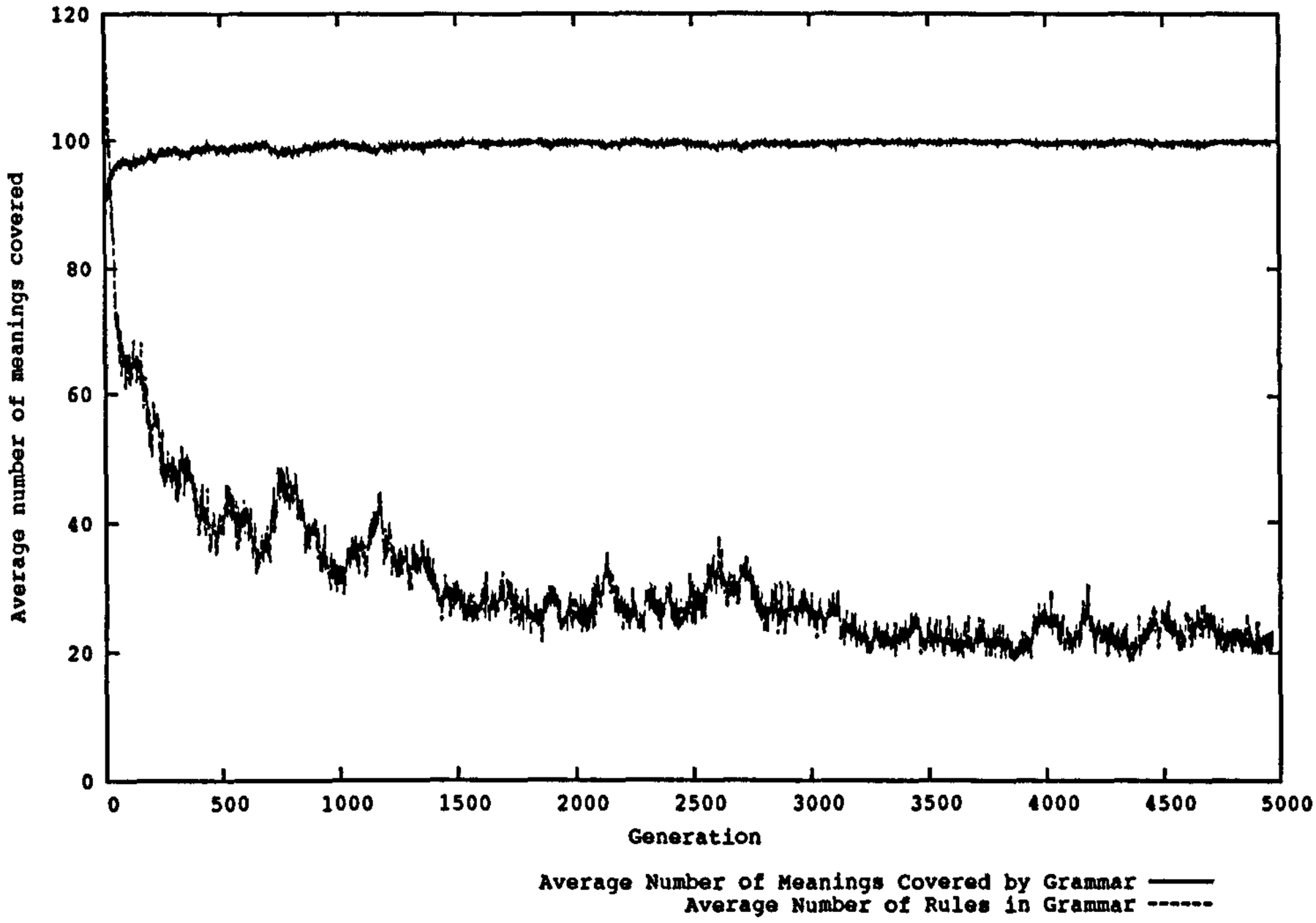

Figure 7.1: *The average number of meanings that can be expressed by agents at each generation of the simulation, and the sizes of their grammars, when using the similarities based inducer but the unaugmented semantic representation.*

that some of the grammars emerging from the simulation are not optimal, as an optimal grammar for the meaning space used would have just 11 rules (one top level rule, plus five "noun" categories and five "verb" categories). Although we would not necessarily expect optimality from every run, unfortunately the new inducer does not appear to perform quite as well as the previous one. However, these results are sufficient to show that compositionality has indeed been achieved. The following is a sample grammar from the final agent of one of the simulations:

s/[P,X,Y] $\longrightarrow$ 3/Y, 3/X, 4/P
3/pete $\longrightarrow$ t
3/mary $\longrightarrow$ y

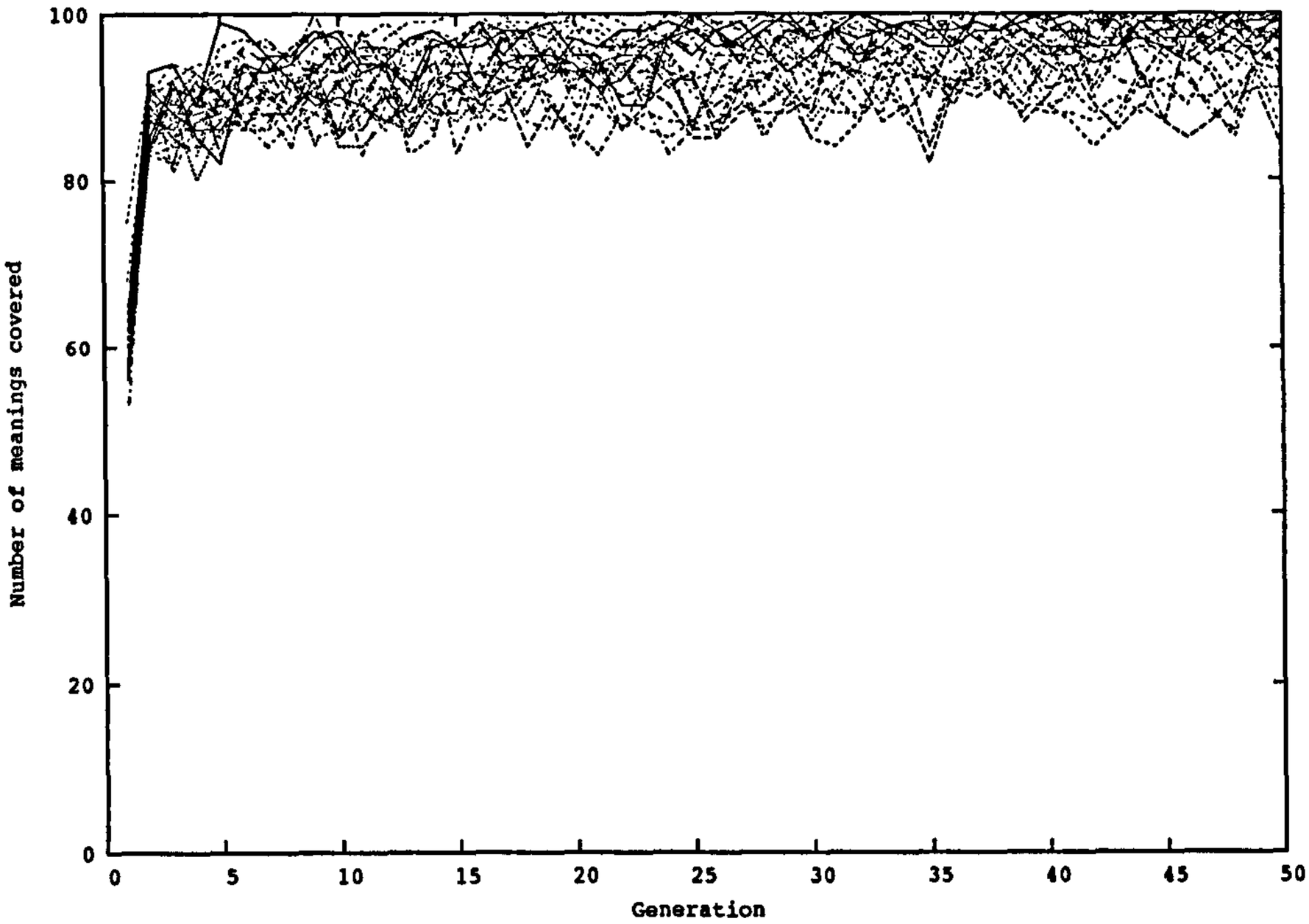

Figure 7.2: The number of meanings that can be expressed by the grammars of agents of each of the first fifty generations in the simulation, when using the similarities based inducer, but the unaugmented semantic representation.

3/john ⟶ a
3/kath ⟶ d
3/anna ⟶ b
4/hates ⟶ o
4/kisses ⟶ g
4/loves ⟶ j
4/adores ⟶ v, p
4/sees ⟶ c

Having established that compositionality can indeed emerge in a system using the new similarities based inducer, the augmented semantic representation was added. Again, runs were 5000 generations in length, with 100 utterances

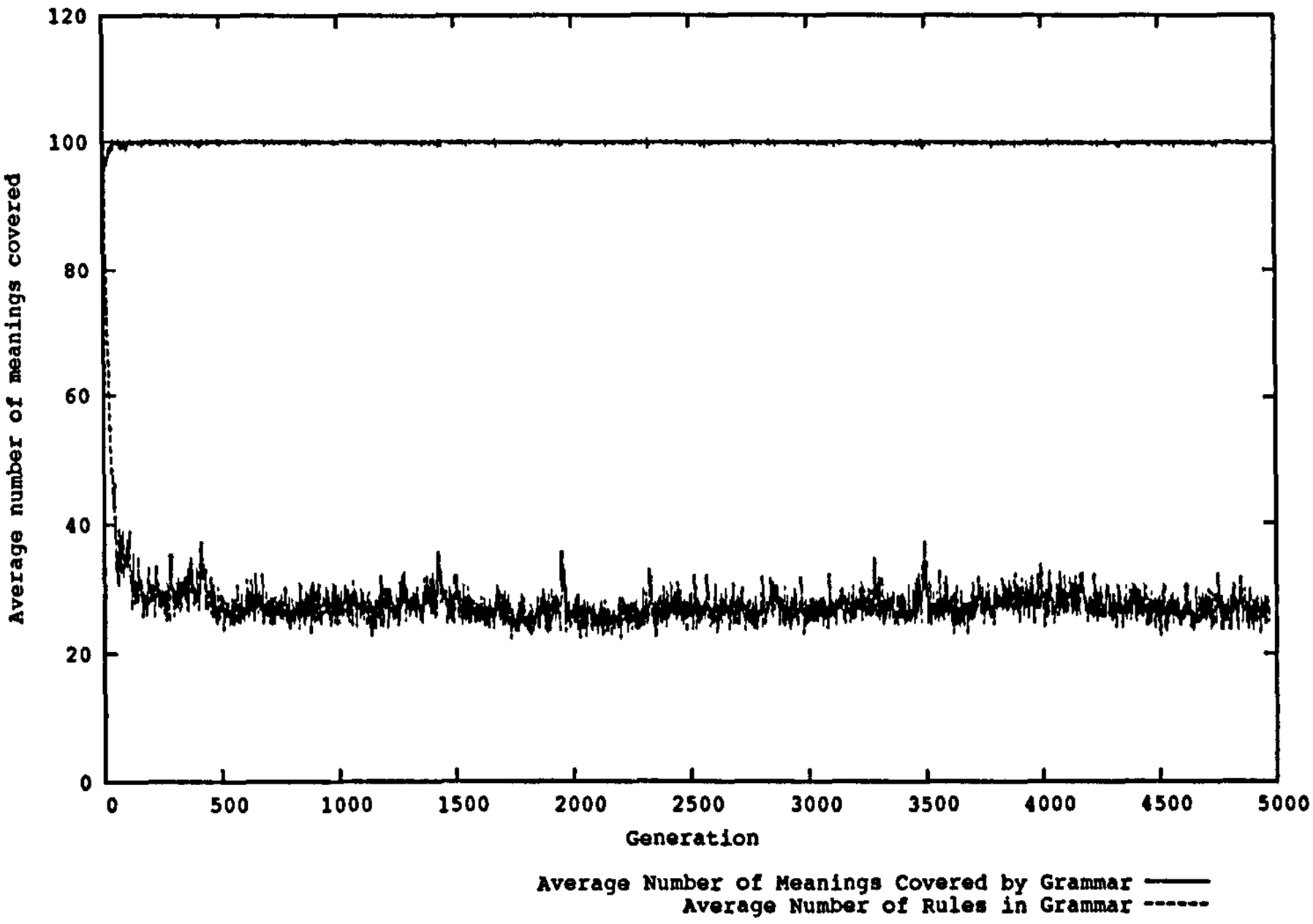

Figure 7.3: *The average number of meanings that can be expressed by and size of grammars for agents of each generation in the simulation,using the similarities based inducer, and the augmented semantic representation.*

per generation. To recap, the augmented semantic representation involves making explicit some information about the "role" of each participant in the event. Thus the meaning [loves,john,mary] would become [[act,loves], [actor,john], [actedon,mary]]. Figure 7.3 shows the proportion of the meaning space that each agent can express, and the size of its grammar, averaged over 26 runs, and figure 7.4 shows the number of meanings agents can express in each of the individual runs for the first 50 generations. As before, a sharp rise in the number of meanings that agents can express can be seen in the first few generations, coupled with a decrease in the size of the grammar due to an increase in the degree of compositionality. What is interesting is that the rise to being able to express almost 100% of the meaning space occurs much more quickly than in the previous experiments, and agents appear to show less deviation from this value than before. Similarly, and even more

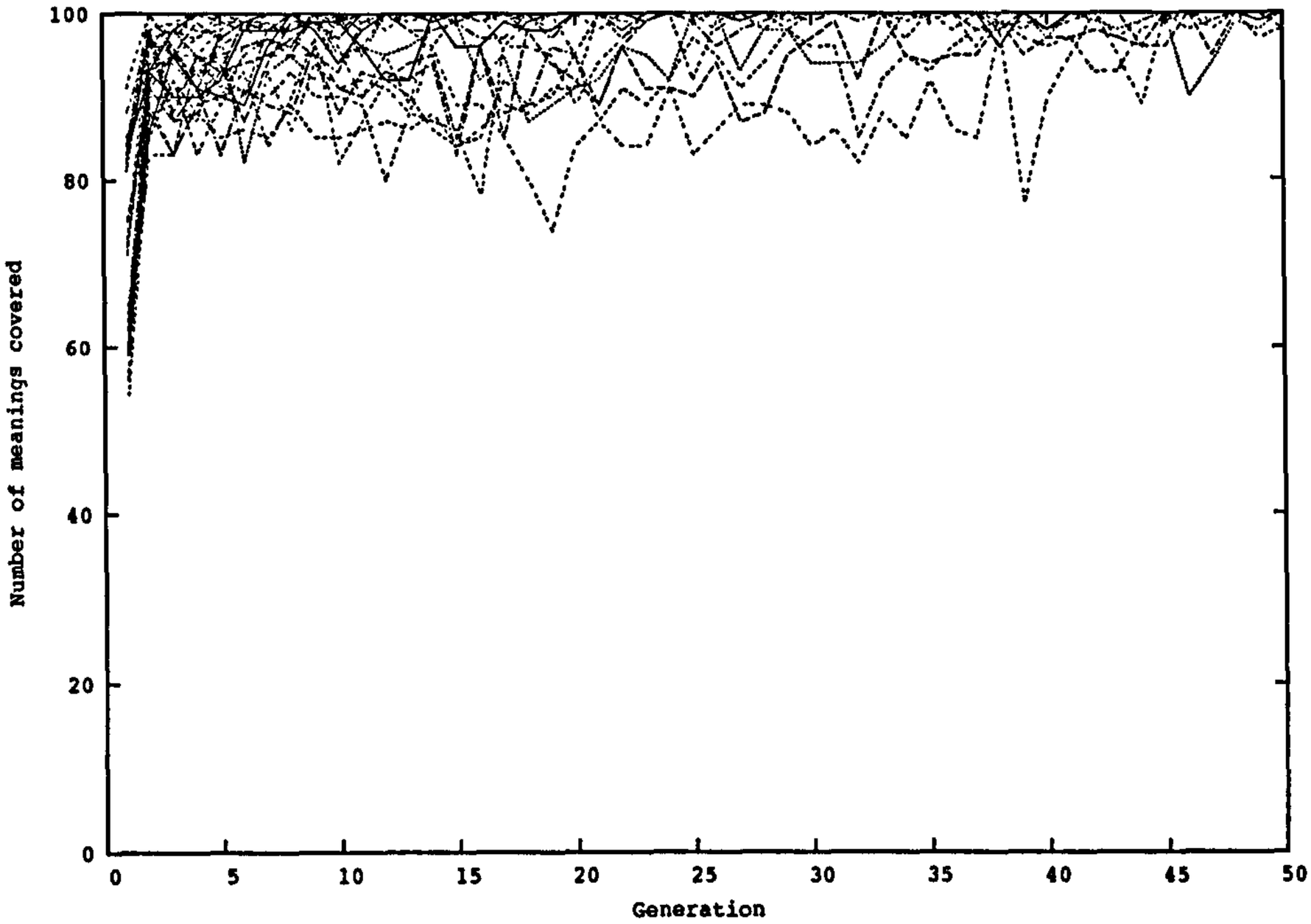

Figure 7.4: The number of meanings that can be expressed by the grammars of agents of each of the first fifty generations in the simulation, using the similarities based inducer, and the augmented semantic representation.

strikingly, the decrease in the size of the grammar down to its minimum value happens exceedingly quickly – well within the first five hundred generations. This time, the average size of grammars is just over twenty – again, reflecting the fact that some of the grammars emerging are not optimal. The optimal grammar type for this representation would have 15 rules: one top level rule, five nouns and five verbs as before, plus one "actor" marker, one "acted-on" marker and two rules to combine these role markers with the noun categories.

But what has happened as a result of including the enriched semantic representation into the system? Certainly "morphemes" representing the explictly represented "role" information were observed, as might perhaps be expected, for this is simply another level of compositionality. How close are the results to a proper case-marked inflectional system, however? Further inspection of

the final grammars reveals that of 26 runs, 17 exhibit case-like behaviour in the final generation, and a further 4 exhibit it in at least one of the final ten generations, with only 5 runs failing to exhibit this behaviour at all. The term "case-like behaviour" is being used to refer to a top level rule that combines a verb and two noun categories, where each of the noun categories is made up of a common noun "stem" plus affixes representing the role information, as in the sample grammar that follows. In this grammar, there is a "subject" marker *p,s*, and an "object" marker *o* which are suffixed on to the noun stems to give nouns of the appropriate categories to build a sentence.

s/[P,X,Y] ⟶ 7/P, 1/X, 3/Y
1/[A,B] ⟶ 9/B, 6/A
3/[C,D] ⟶ 9/D, 5/C
6/actor ⟶ p, s
5/actedon ⟶ o
9/anna ⟶ e, v
9/kath ⟶ n, s
9/pete ⟶ s
9/mary ⟶ u
9/john ⟶ b, h, s
7/[E, loves] ⟶ 11/E, k, r, e
7/[F, kisses] ⟶ 11/F, x
7/[G, sees] ⟶ 11/G, r, r, q
7/[H, adores] ⟶ 11/H, l, g
7/[action, hates] ⟶ z, n
11/action ⟶ k

In this particular case, a common affix "k" has been identified as signalling the role of the *action* in 4 out of the 5 predicates in the meaning space.

Those simulations which did not converge on grammars incorporating in-

flectional endings tended to exhibit one of two types of behaviour (or some combination of the two):

- The first is two entirely distinct noun categories, one used to express the subject of the sentence, and one used to express the object, as in the grammar fragment that follows. It can be seen that each of the two "versions" of the same noun are completely unrelated, i.e. there is no common noun stem which denotes the individual being referred to, and similarly nouns of the same class have nothing in common: there are no subject and object markers. This is essentially a grammar of the we called "Type B" in Chapters 5 and 6.[6]

  s/[P,X,Y] ⟶ 3/P, 6/X, 1/Y
  6/[agent, john] ⟶ e, f, b
  6/[agent, kath] ⟶ o, v, u, x
  6/[agent, anna] ⟶ a
  1/[patient, kath] ⟶ r, e
  1/[patient, john] ⟶ g
  1/[patient, anna] ⟶ h
  3/[act, loves] ⟶ q
  3/[act, sees] ⟶ w
  3/[act, hates] ⟶ y

- The second type of grammar *does* exhibit inflectional case markings to denote the subject and object of a sentence; however, the noun *stems* to which they are affixed are unrelated. For example in the grammar fragment that follows, the suffix *i* denotes the subject of the sentence, and when combined with the string *v,s* will denote the individual "john"

[6]Note that the emergence of two distinct noun categories is now guaranteed by the augmented semantic representation, which renders rules for subject and ojbect nouns unmergeable.

in the role of event-perpetrator, whilst the prefix *h* signifies the object. But in order to represent "john" as the acted-on party in the event being described, this string is combined not with *v,s* as before but with a different string.

s/[P,X,Y] ⟶ 1/Y, 13/X, 6/P
13/[A,B] ⟶ 16/B, 15/A
1/[C,D] ⟶ 3/C, 8/D
15/actor ⟶ i
3/actedon ⟶ h
16/john ⟶ v, s
16/jane ⟶ k, h
8/john ⟶ e, v, s, n
8/jane ⟶ s
6/[E,F] ⟶ 14/F, 7/E
7/act ⟶ i, b
14/adores ⟶ y, n, k
14/loves ⟶ j, v
14/sees ⟶ h, m, l

It is interesting to note also that in this grammar the ordering of stem and affix is inconsistent: for the subject of the sentence, the stem occurs before the inflectional marking, whilst for the object it occurs after it. This is a common, though by no means universal, feature of the grammars emerging from these simulations, and happens as a result of the fact that there is nothing within the bias of the learner or its innate knowledge of the language to constrain it to a common stem-affix ordering. Clearly this is one aspect of case grammars in natural language that must be explained.

Thus we have established that inflectional behaviour is common in the out-

put of the system using the new similarities based inducer in conjunction with the augmented semantic representation. However, it would appear that such grammars are not as stable, or not as reliably passed on from one generation to the next, as those which form the output of the original system. Grammars of this type appear to formed and lost again many times during the 5000 generations of a single simulation, often remarkably quickly, hence the 4 runs which exhibit case marking at some point during their final 10 generations, but not in the final one. This could simply be attributable to the fact that the augmented semantic representation makes sentences harder to learn, effectively dividing them up into six segments instead of only three.

Another possible reason for the instability exhibited is the tendency of different agents to *segment* the string in different places according to the order in which utterances are presented, as described in Section 7.3.1. Returning to the "toy" inflectional language presented in Section 7.1, in which the utterance [[act,loves], [actor,john], [actedon,mary]] is represented by the string *j,o,h,n,a,l,o,v,e,s,m,a,r,y,b*, we can see that the correct attribution of inflectional markers to roles is crucially dependent on the order of sentence presentation. For example, if the next utterance presented were [[act,hates], [actor,john], [actedon,kath]], represented by the string *j,o,h,n,a,h,a,t,e,s,k,a,t,h,b*, then the substring *j,o,h,n,a* would be attributed to the meaning element [actor,john], and the affix *a* will (ultimately) be correctly identified as an inflectional marker denoting the role of the participant *John*. However, if the next utterance were instead [[act,loves], [actor,pete], [actedon,kath]], represented by the string *p,e,t,e,a,l,o,v,e,s,k,a,t,h,b* then the result would be entirely different: this time, the common semantic element [act,loves] will be attributed to the common substring *a,l,o,v,e,s*, thus transposing the inflectional marker *a* from a suffix of the noun to a prefix of the verb.

Thus, once a grammar has emerged, the apparent "meaning" of affixes etc. can change dramatically in the process of transmission of the language, although the actual utterances produced by each grammar for a give meaning

should be the same. However, during the actual *emergence* of an incomplete grammar, this could have a dramatic effect on those utterances which are produced by invoking the *invention* algorithm, and this might possibly be sufficient to prevent the grammar from ever stabilising fully. A fuller investigation of the stability of grammars emerging from the current set-up would be needed to establish whether or not this is indeed the case. However, the current results have amply demonstrated that the use of a similarities based inducer and a richer semantic representation is sufficient to simulate the emergence of a primitive inflectional system of case markings within an Iterated Learning Model type paradigm.

What also seems to be apparent is a lack of consistency in stem-affix ordering: in some grammars the stem and affix will have the same ordering for both subject and object constructions, and in others they will differ. This is unsurprising, because the original stem-affix distinction has to be made on the basis of chance similarities between strings. If these similarities happen to be between noun stems, then consistent ordering of stem and affix is automatically guaranteed; however if a similarity between affixes is found before the stem has been identified, there is nothing to prevent a similar further chance similarity from occurring for the affix of the other type in which it has the alternative position with respect to the noun stem.

7.5 Modifying the bottleneck

Having established that inflectional behaviour is common in the output of simulations using the new similarities based inducer in conjunction with the augmented semantic representation, and based on the observation that such inflection is really just another layer of compositionality, once again, experiments were performed in which the transmission bottleneck was externally manipulated to see if tightening it would increase the occurrence of the types

of behaviour we are seeking here.

As in previous experiments in Sections 5.4 and 6.4, the number of utterances per generation was once again increased from 100 to 1000, and the bottleneck was controlled by allowing agents to select meanings from only a randomly chosen subset of the meaning space of predetermined size. Each agent is allowed a different subset of the meaning space to use, therefore when an agent makes the transition from learner to speaker, the utterances that it will be called upon to produce will be a different set to those to which it was exposed when it was a learner. This set could well include meanings to which it has had *no* previous exposure, and which may not even be covered by the grammar it has learned.

Once again, a variety of subset sizes were tried, ranging from 20% to 100% of the meaning space. The table below shows the proportion of simulations that resulted in case-like behaviour in their final generation (of 5000). The proportion of additional simulations which showed this behaviour in at least one generation of their final 10 generations is also given, as well as those that failed to show it at all:

subset size	percentage exhibiting case-like behaviour		
	in the final generation	in the final 10 generations	in none of these
30%	30.77	7.69	61.54
40%	48.00	32.00	20.00
50%	64.00	36.00	0.00
60%	69.23	15.38	15.38
70%	58.33	16.67	25.00
80%	46.15	23.08	30.77
90%	32.00	36.00	32.00

Figure 7.5 is a bar chart showing both the proportion of simulations which

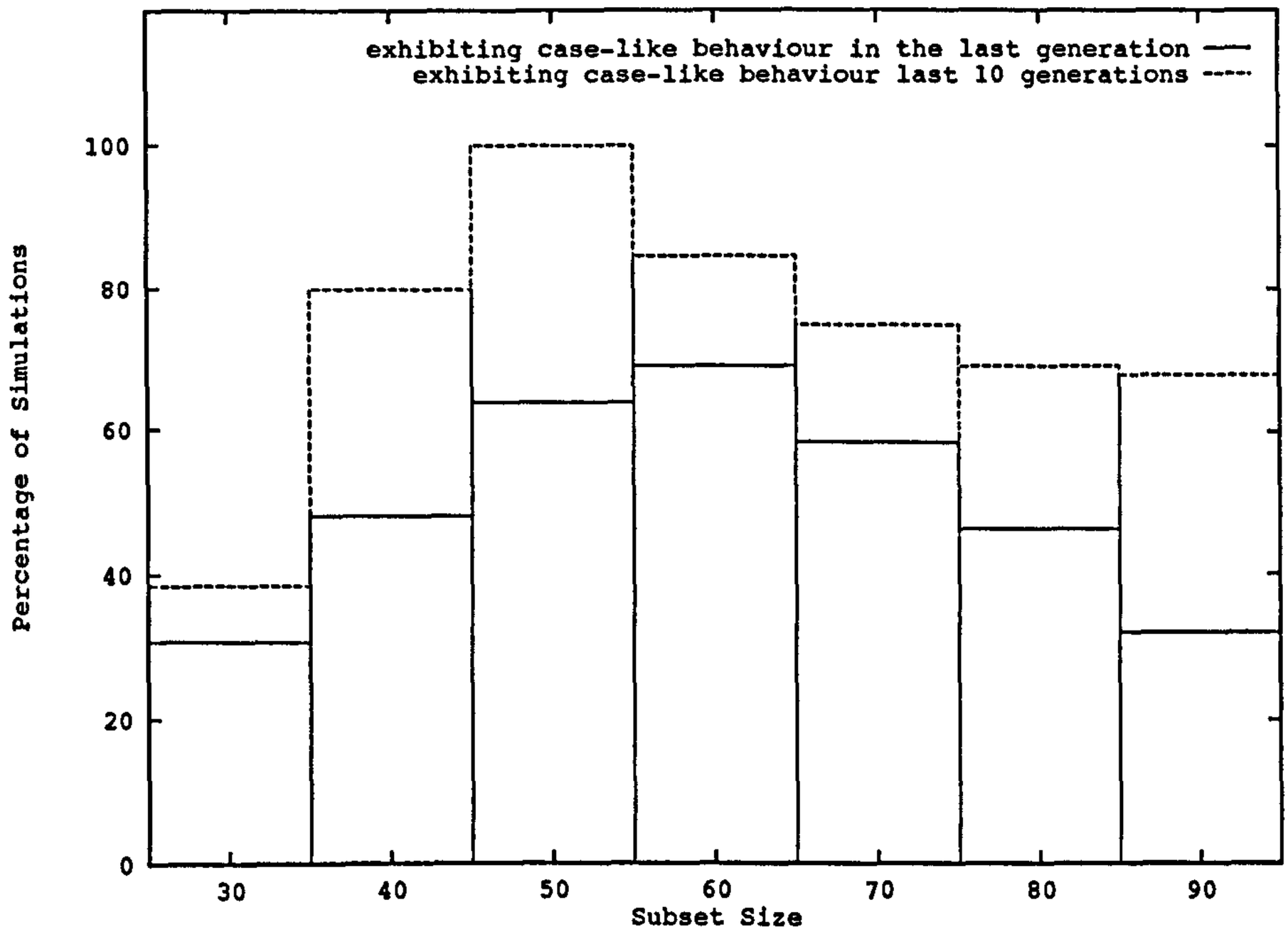

Figure 7.5: *The proportions of simulations which exhibit case-like behaviour in the final (5000th) generation, and in also in the at least one of their final ten generations.*

exhibit case-like behaviour in their final generation, and also those that show it in at least one of their final ten generations. What is clear from both the graph and the table above is that the emergence of case-like behaviour seems to be most strongly favoured when agents select meanings from subsets of between 50% and 60% of the entire meaning space. This is interesting because it represents a much more relaxed bottleneck than that which is optimal for the emergence of compositionality in the simple case where we are not trying to achieve inflection. This is perhaps to be expected – the degree of compositionality we are trying to achieve here is more complex than that encountered previously, and thus it follows that agents might require exposure to a larger proportion of the meaning space in order to make effective generalisations regarding how the meaning of a string may relate to the meaning of its parts.

Also worthy of note is that even with an optimal bottleneck, there is not a great deal of improvement seen on the results of the experiments described in Section 7.4, where agents only make 100 utterances per generation but have access to the entire meaning space at all times: recall that under those circumstance, 17 out of 26 simulations (65.38%) were found to exhibit case-like behaviour at the end of 5000 generations, and a further 4 (15.38%) showed this behaviour in at least one of their final ten generations. These are very similar figures to those observed with a subset size of 60% above. The only subset size which shows any improvement on this is for 50%, where the proportion of simulations exhibiting case-like behaviour in their last generation is ever so slightly less, but where all other simulations show this behaviour in at least one of their final ten generations: there are none that fail completely to exhibit it.

The fact that the emergence of a basic compositional grammar, and the emergence of case-like behaviour are favoured by different bottleneck sizes suggests an interesting possibility for a way forward: perhaps *altering* the bottleneck during learning would prove fruitful. Agents would start life with a vary narrow bottleneck which might aid in the acquisition of basic compositionality, and this could later be relaxed to facilitate the acquisition of case. This idea is clearly related to Elman's "starting small hypothesis" [31]: an agent's experience of language is limited during its early life to only simple constructions. This is achieved by limiting its working memory and then gradually allowing it to increase as the agent matures, thus allowing it to turn its attention to more complex linguistic phenomena. This has been found to greatly facilitate the acquisition of a grammar. It also seems that it might be a plausible feature of a child's learning environment: when children are very young it seems likely that topics of conversation to which they are exposed might be quite limited and that this may well broaden as they mature and start to find themselves in a wider range of differents scenarios.

7.6 Discussion

In this chapter, we have attempted to demonstrate the emergence of an inflectional system of case marking within an Iterated Learning Model type system. This system was based strongly on Kirby's [44] original model but incorporated two notable changes:

- Firstly, the induction algorithm used by the model was modified so that it operates on the basis of maximal similarities rather than maximal differences.
- Secondly, the semantic representation employed was augmented so as to make explicit information that was previously only implied by the positioning of elements in the vector meaning.

7.6.1 How plausible is the similarities based inducer?

In his discussion over the choice of learning algorithm employed in [44], Kirby states that the difference-based inducer was chosen primarily for efficiency and ease of analysis. He emphasises that no claim is made as to its efficacy as a practical grammar induction tool. Is it reasonable to use an inducer that works on the basis of maximal similarities rather than differences as we have done here? Is there any evidence to suggest that early human language learners were equipped with either algorithm?

As discussed in Chapter 2, Alison Wray [88, 89], argues that early language may have been "holistic", made up of phonetically arbitrary utterances, in which there was no phonological similarity between sequences with similar meanings. Such a system would have contained sufficient and necessary utterances for everyday communication, and thus would have been very stable.

Bickerton [7] has suggested the existence of a non-language specific cognitive module responsible for keeping track of thematic roles (the "who did what to whom") which he suggests might have arisen in pre-linguistic species to perform some kind of "social calculus". According to Bickerton's account of language evolution, this module has been exapted to assign theta roles to pre-existing referential vocabulary items, leading to order relationships between referents and the action word they are combined with. Wray proposes that in her hypothetical holistic proto-language there are no *words* to which roles might be applied, thus the theta-role module instead tries to *find* them by segmenting the strings. This, so far, is in-keeping with both the difference based inducer used by Kirby [44] and here in Chapters 5 and 6 and the similarities based inducer described in the current chapter. So which method might these hypothetical proto-language speakers have favoured? Wray herself suggests similarities: "if in two or more sequences, there were chance matches between phonetic segments and aspects of meaning, then it would seem as if there was a constituent with that meaning" [89]. Further support for this idea can be found in the literature on child first language acquisition.

Ann Peters [59] proposes that the initial units of language acquistion may include a significant number of larger phrases which are initially learnt by rote and unanalysed, but which will eventually be segmented into short word-length units. Whilst she believes that children primarily segment these larger units into smaller ones on the basis of phonological salience, e.g. by segmenting off the first/last/stressed syllable from the rest, or by segmenting at rhythmically or intonationally salient places, she does go on to suggest that once this segmentation has occurred, comparison *between* units may occur. In particular, the subdivision into *frames* and *slots*: "if two (or more) units, after segmentation ... [on the basis described above] ... appear to share a common subunit, A, followed or preceeded by alternative subunits, B and C ... take note of this fact" (pg48). This would appear to have characteristics in common with our new *findsimchunks* heuristic. Furthermore,

she claims that such *frames* can be generalised from repeated instances of particular constructions, and can then be used as a further segmentation aid: as templates for the segmentation of newly heard utterances.

This is only one of many methods that she suggests language learners may employ to break up these longer linguistic units into their constituent parts, but it demonstrates that a similarities based inducer such as the one presented in this chapter is at least in keeping with the data on child-language acquisition.

7.6.2 What does the augmented semantic representation actually capture?

As previously mentioned, it is important to stress here that we have not actually *added* any information with this change, only made explicit what was previously implicit. However, what was that implicit information? Are we directly specifying the "subject" and "object" of a sentence? And does this mean that in order to evolve such grammars, agents need innate knowledge of these syntactic categories? Or could it be that we are really marking an inherent property of individuals themselves and the way they take part in the interactions being described. One naive way to look at it might be as some kind of "thematic role". However, this is not really adequate, as the categories we are using are applied universally to all subjects and to all objects of transitive verbs, regardless of the role types required by these verbs. For example, the verb kiss would usually be viewed as selecting an agent as its subject and a patient as its object, whilst a stative verb such as *love* is generally considered to have an experiencer in the subject position.

Dowty [29] claims that the traditional view of thematic roles as a set of finite discrete categories is inadequate, and argues instead for proto-role types to

which arguments will have different degrees of membership. Thus only two categories are needed, the *proto-agent* category and the *proto-patient* category. Each has an associated set of contributing properties (e.g. volitional involvement, sentience for proto-agents; undergoing a change of state, being causally affected for proto-patients), and subject and object of a sentence are selected according to the number of these contributing properties each argument possesses: the argument with the greatest number of proto-agent characteristics becomes the subject of the sentence, and of those remaining, the one with the greatest number of proto-patient characteristics becomes the object, and if there are further arguments, the one with the next greatest number of proto-patient characteristics becomes the indirect object.

Thus if we consider our "actor" and "acted-on" categories to denote *proto-roles* such as these, we can get around having to make the concepts of *subject* and *object* explicitly available to learners: it will be possible to determine which role each argument has purely from the semantics of the event taking place, and no innate knowledge of syntactic categories will be required.

7.7 Summary

In this chapter, we have made changes to the original Iterated Learning Model implemented in Chapter 4 and based on Kirby [44] in an attempt to simulate the emergence of inflectional case markings. We have successfully demonstrated that it is possible to obtain such structure in a population of agents without any innate language specific knowledge, purely resulting from the biases of the learners and the dynamics of language transmission. This is perhaps not entirely unsurprising, because what we are looking to achieve is really no more than another level of compositionality. However, the inflection that emerged is somewhat sub-optimal: agents seem unable to reliably determine to which element of the sentence an affix belongs, and there is no

guarantee of consistency of stem-affix ordering, and these respects it differs quite noticeably from the case systems which occur in natural languages. Attempts to promote the emergence of case-like behaviour by external manipulations of the bottleneck of language transmission have shown that, like the other experiments described in previous chapters, case seems most likely to emerge in the presence of moderate sized bottlenecks. However, only a slight improvement over the condition where bottlenecks were not manipulated was seen.

Chapter 8

Conclusions

In this thesis, attempts have been made to simulate the emergence of case-like behaviour to distinguish thematic roles in populations of communicating software agents, based on Kirby's Iterated Learning Model [44]. This undertaking was based on the observation that the outcome of Kirby's original simulations can be likened to the "proto-language" stage of Jackendoff's theory of incremental language evolution. That is, they result in populations of agents putting together meaningful symbols to form larger utterances whose meanings are a function of the meanings of the constituent symbols. The key elements of his proto-language stage are all present (open vocabulary, concatenation of symbols, use of basic word order to express semantic roles) and the simulations also seem to show evidence of the emergence of syntactic categories, thus fulfilling the prerequisites, according to Jackendoff's hypothesis, for the emergence of case.

8.1 Overview of Results

In Chapter 1 we introduced the notion of the Iterated Learning Model [46] (described in more detail in Chapters 3 and 4) and how it has been used to demonstrate the emergence of compositional and recursive syntax in populations of agents with no explicit knowledge of the nature of language. This behaviour has been attributed to the interaction of the biases of the learner with the dynamics of language transmission, namely the learning bottleneck. Under circumstances where agents cannot hope to hear all possible utterances from a language during their lifetimes, languages in which the meaning of an utterance can be predicted from the meaning of its parts will have a strong selective advantage over a purely holistic language where there is no relationship between meaning and form. This notion is persuasive, but how great is its explanatory power? Can it be used to illustrate the emergence of other features of language? It is notable that in the results of these simulations, semantic relationships are universally signalled using word order. None of the languages emerging exhibit any of the properties of case grammar. This is clearly a deficit as it does not reflect the true nature of human language. Thus attempts were made to investigate whether case-grammars were an equally plausible outcome of this kind of simulation.

Firstly, in Chapters 5 and 6, attempts were made to create a selective pressure for the emergence of grammars containing two distinct noun categories, one used to express the subject of the sentence and one the object, using variation in word order. These grammars had already been observed to occur spontaneously in the implementation of Kirby's Iterated Learning Model described in Chapter 4. It was hoped that the occasional reordering of elements of a sentence resulting in the presence of conflicting word orders in which subject and object have been inverted would result in linguistic selection for distinguishable subject and object noun categories. Certainly, the changes introduced in Chapter 5 did indeed seem to increase the frequency of

occurrence of such grammars; however it was noted that simply changing the parsing and production algorithm used within the simulation was sufficient to cause a significant swing towards the behaviour being sought. Adding variability of word order to this increased its effect further, but the change seen was actually smaller in magnitude than that created by the change in production algorithm.

Based on the observation that the learning bottleneck is crucial to the emergence of language-like behaviour, attempts were made to reinforce the results of these experiments by externally manipulating the size of this bottleneck. This was first implemented without any freedom of word order, under which circumstances, it can be seen that a tight bottleneck seems to indeed result in a greater drive towards compositional behaviour, and optimal grammars of Type A, containing a single noun category used to express both subject and object of the sentence. However, as the bottleneck is relaxed an interesting trend can be seen: as might be expected, the chances of a simulation converging on a grammar that can be classified into either group starts to decrease, as does the likelihood that any grammar arrived at will display optimal characteristics; however, the notable corollary to this is the increase in the proportion of Type B grammars emerging relative to those of Type A, peaking at bottlenecks of between 50% and 90% of the meaning space. A similar effect is seen when occasional sentence reordering as discussed in Chapter 5 is re-introduced. Again, tighter bottlenecks favour Type A grammars, and as the bottleneck is relaxed, the quality and regularity of grammars decreases, in conjunction with the emergence of an increased proportion of Type B grammars, which seems to peak when approximately 40% of the meaning space is in use.

Thus it was shown in Chapter 5 that the introduction of word order freedom does indeed seem to result in the emergence of a higher proportion of grammars of Type B. However, what is also clear is that this behaviour can also be promoted by other means which would not be anticipated to cre-

ate a direct pressure for distinguishable subject and object forms, such as the introduction of a non-deterministic parsing and production algorithm, or manipulation of the transmission bottleneck. Furthermore, it seems that the mechanism by which the presence of alternative word orders results in the increased frequency of Type B grammars may not be as a result of pressure for distinguishable subject and object noun forms that was anticipated: because agents are not required to correctly interpret each other's utterances, there is actually no need to disambiguate conflicting word orders. And because agents will not seek to associate more than one meaning with a given string, on the whole these word orders can *only* be propagated in grammars which *already* have distinguishable subject and object categories. Ultimately then, the increase in the incidence of Type B grammars achieved due to the introduction of alternative word orders was deemed to be artefactual, perhaps the result of other factors such as the general disruption caused by these interventions, resulting in the greater emergence of suboptimal grammars.

In Chapter 6, attempts were made to address the fact that there is no requirement for agents to *understand* each other, by making the learner agent intolerant of utterances which appear to mean something other than what the speaker intended. That is, if the learner parses the utterance and the wrong meaning is returned, the string is rejected, and the speaker must try again. Although this alone is not effective, when combined with a moderate sized penalty applied to the grammar rules used to produce the misinterpreted utterance, this does indeed appear to result in some pressure for the emergence of grammars with distinguishable subject and object. However, the rejection of ambiguous utterances is also highly de-stabilising, resulting in grammars that are far more irregular, far less likely to fully converge on a compositional grammar of either type, and which are inconsistent in the role applied to each of the noun categories. Therefore, although this approach does result in an increase in the number of grammars emerging with two noun categories relative to the number with just one, the absolute numbers

are still lower than before the freedom of word order and rejection/penalty were introduced.

As in the previous chapter, the effect of manipulating the size of the bottleneck was again investigated to see if, by creating an increased pressure for compositionality, this would help counteract the disruptive effects of the use of penalties. Again, a trend was seen whereby Type A grammars are favoured by tighter bottlenecks, and as the bottleneck is relaxed, the proportion of Type B grammars emerging relative to those to those of Type A increases. This time the effect was much more dramatic, such that for a bottleneck size of 60%, there were *no* Type A grammars seen at all, and a massive 88.89% of grammars that did emerge were of Type B. However, a caveat to this is the fact that an extremely large proportion of the simulations that were run had to be terminated early due to problems experienced elsewhere with increasing rule length and grammar size.

Finally, in Chapter 7 a different approach was introduced. Rather than assuming that the use of case in primitive natural language might have emerged in response to a need to disambiguate subject and object of a sentence due to use of multiple word orders, it is suggested that case may actually have emerged first, perhaps for some other other linguistic purpose, and that it was only after this that multiple word orders *became* a linguistic possibility. Thus we set about trying to generate proper inflectional case markings in the absence of free word order. This essentially involved making changes to the system such that it was able to view inflectional endings as simply another level of compositionality. Here some success was achieved, in that grammars that exhibited a common noun stem plus inflectional affixes specifying the role of the participant in the event being described did emerge. However, this behaviour was not consistent – sometimes there were no inflectional endings at all, and in other cases, the endings emerged but different "stems" were required when the subject and object of a sentence was being expressed. There was also a problem with the interpretation of which word an affix belongs

to: what is intended as a suffix on the subject might easily be interpreted as a prefix on the verb in a language where the verb follows the subject. Furthermore, grammars were often inconsistent in their stem affix ordering. Clearly, these issues are intimately related: if agents were constrained to expect a consistent ordering of stem and affix then both issues would be resolved – agents would simply have to determine whether the language they are learning uses prefixes or suffixes.

When experiments with bottleneck manipulation were carried out in this context, it was found that moderate bottlenecks produced the optimal results: for both very tight or very relaxed bottlenecks, the incidence of case-like inflectional endings was much lower. Again, the optimal value seemed to be between 50 and 60%. However, this did not represent a great improvement on the situation where the bottleneck was left untouched.

Thus we must conclude that the evolution of case-like behaviour in a system like Kirby's iterated learning model is a non-trivial problem. On the whole, attempts to introduce pressures for the emergence of case are also disruptive to the emergence of compostional syntax itself. This is perhaps due to deficiencies in the model, both in the implementation of rule scoring and penalising developed here, and also the known problem with the induction algorithm and its tendency to leave semantically redundant non-terminal characters stranded in top-level rules, which can lead to escalating string length and grammar size [71]. When attempts were made to resolve some of these disruptions, in the form of experiments manipulating the transmission bottleneck, the interesting result was that non case-like grammars seemed to be favoured by very tight bottlenecks, whilst slightly more relaxed bottlenecks seem to favour grammars with case-like properties (that is distinguishable subject and object categories).

It is argued that this is because, from a compositional point of view, the types of grammar we are aiming to achieve are suboptimal: despite being

non-holistic in the sense that the meaning of utterance is composed from the meaning of its parts, these grammars will require agents to observe a greater proportion of the meaning space in order that they can be faithfully transmitted from one generation to the next than their counterparts which use the same form of the noun for both subject and object. Thus when the bottleneck is very tight, resulting in a very high pressure towards compositionality, then these simpler grammars will be favoured, regardless of what pressure there may be in the environment for distinguishable subject and object categories. After all, distinguishable subjects and objects are of no use to a language that cannot be passed from one generation to the next accurately. Or more specifically, even if distinguishable subjects and objects should arise in a language emerging under conditions where there is a very tight bottleneck, they are likely not to be accurately transmitted from one generation to the next, and will thus quickly be replaced by the common or garden kind of noun category that can be used in any semantic role. (This would presumably itself exert a pressure for word order to become fixed, although this would not be a visible outcome of the studies carried out here, because freedom of word order was externally imposed).

However, once the transmission bottleneck is relaxed a little, languages are freed from the constraint of needing to be as compositional as possible, and this allows other pressures to start pushing them in other directions. Thus, weaker bottlenecks may favour case-like languages because the selective pressure for compositionality they exert is no longer stronger than the need to be able to disambiguate in situations where word order cannot be relied upon to make semantic relationships explicit.

One question that must be posed is whether a similar interaction of antagonistic pressures could be at work in human languages and whether this could in part be an explanation for why some languages exhibit case whilst others do not, or at the very least why case is sometimes lost, for example the transition made from strongly case-marked old English to the almost case-less

language that is contemporary English. It might also help explain why pidgin and creole type languages tend to exhibit rigid word order [5] and other such phenomena that have led many researchers to believe that word order has primacy as a mechanism for signalling semantic relationships – a notion that is contested in some corners.

8.2 Future work

One important area for future study would be attempting to resolve the known limitations of the current model, with regard to problems in the parsing algorithm relating to the rewarding and penalising of rules, and also the issues with repeated semantically null sequences. If a different learning algorithm were used, e.g. that employed by Vogt [85] in his work on iterated learning in grounded agents, would the innovations designed to introduce pressure for distinguishable subject and object categories still prove disruptive to the emergence of compositionality? This is an important question: to what degree are the results seen here implementation specific?

Another important avenue to pursue is the possibility that case did not evolve in order to aid disambiguation of subject and object categories where word order cues are unreliable, but in fact for some other purpose, such as to facilitate the acquisition of word orders that would otherwise prove difficult for sequential learning devices such as our human brains to master. Clearly Kirby's learners are not sequential learning devices, and word order considerations have no relevance to them, but it would be informative to experiment with such learners in an iterated learning setting to see if case spontaneously evolves in languages with those word orders that are more difficult to acquire. It would also be of interest to pursue further experimentation with weak and strong bottlenecks in this context to examine whether the idea that strong bottlenecks favour no case and rigid word order, whilst weaker ones will allow

pressures that might result in case-grammar to come into play.

Finally, another profitable avenue for future work in this area would involve grounded agents: it is clear from Vogt's [85] and Steels' [79] studies with situated agents (or simulations thereof) that shared contexts and the development of representations of objects and events can have a huge influence on the development of linguistic behaviour, as in Steels' agents' construction of syntactic categories, and the superior performance seen in "guessing games" over "observation games" in many aspects of the emergence of compositionality in Vogt's work.

8.3 Conclusion

Experiments have been presented in which attempts were made to demonstrate the emergence of inflectional case markings in populations of software agents. The model used was based on Kirby's Iterated Learning Model [44], with modifications to enable case-like behaviour to emerge. As in Kirby's model, agents have no innate knowledge about language or its structure. It was discovered that the use of word order freedom to as a selective pressure for case-like behaviour is not very effective. However, treating case as simply another level of compositionality does result in some degree of inflectional behaviour, but it tends to be irregular and not very reliable. Furthermore, experimenting with the language bottleneck has returned the interesting result that very restrictive bottlenecks seem to select for case-less grammars, whilst slightly looser ones seem to favour case-like behaviour in the presence of word order freedom. This is attributed to the fact that when the bottleneck is very tight, this will exert a strong pressure for compositionality, over and above any pressure for disambiguation that might result from variability in word order. Only when the pressure for compositionality is relaxed slightly will the sub-optimal (from a compositional point of view) grammars

with distinguishable subject and object categories start to emerge. It is postulated that this may be a factor in the development of case also, or at least a possible explanation as to why languages sometimes lose their case-structure in favour of strict word order as in the transition from Old to contemporary English.

Bibliography

[1] Jean Aitchison. *The Seeds of Speech.* Cambridge University Press, 1996.

[2] Jean Aitchison. *The Articulate Mammal.* Routledge, fourth edition, 1998.

[3] John Batali. Computational simulations of the emergence of grammar. In James Hurford, Chris Knight, and Michael Studdert-Kennedy, editors, *Approaches to the Evolution of Language: Social and Cognitive Bases.* Cambridge University Press, 1998.

[4] Elizabeth Bates, Brian MacWhinney, Cristina Caselli, Antonella Devescovi, Francesco Natale, and Valeria Venza. A cross-linguistic study of the development of sentence interpretation strategies. *Child Development*, 55:341–354, 1984.

[5] Derek Bickerton. *Language and Species.* University of Chicago Press, 1990.

[6] Derek Bickerton. Catastrophic evolution: the case for a single step from protolanguage to full human language. In J. R. Hurford, M. Studdert-Kennedy, and C. Knight, editors, *Approaches to the Evolution of Language: Social and Cognitive Bases.* Cambridge University Press, 1998.

[7] Derek Bickerton. How protolanguage became language. In Chris Knight, Michael Studdert-Kennedy, and James Hurford, editors, *The Evolution-*

ary Emergence of Language: Social Function and the Origins of Linguistic Form. Cambridge University Press, 2000.

[8] M. D. S. Braine. The acquisition of language in infant and child. In C. E. Reed, editor, *The Learning of Language*. New York: Appleton-Century-Crofts, 1971.

[9] Henry Brighton. Compositional syntax from cultural transmission. *Artificial Life*, 8(1):25–54, 2002.

[10] Henry Brighton. *Simplicity as a Driving Force in Linguistic Evolution*. PhD thesis, Department of Linguistics, University of Edinburgh, 2003.

[11] Henry Brighton and Simon Kirby. The survival of the smallest: Stability conditions for the cultural evolution of compositional language. In J. Keleman and P. Sosik, editors, *Advances in Artificial Life*. Springer, 2001.

[12] Henry Brighton and Simon Kirby. Meaning space structure determines the stability of culturally evolved compositional language. Technical report, Language Evolution and Computation Research Unit, University of Edinburgh, 2002.

[13] Ted Briscoe. Language as a complex adaptive system: Coevolution of language and of the language acquisition device. In *Proceedings of the 8th Computational Linguistics in the Netherlands Conference*, 1998.

[14] Ted Briscoe. The acquisition of grammar in an evolving population of language agents. In Stephen Muggleton, editor, *Machine Intelligence 16*, 1999.

[15] Ted Briscoe. Grammatical acquisition and linguistic selection. In E. J. Briscoe, editor, *Linguistic Evolution through Language Acquisition: Formal and Computational Models*. Cambridge University Press, 1999.

[16] Ted Briscoe. Grammatical acquisition: Inductive bias and coevolution of language and the language acquisition device. *Language*, 76.2, 2000.

[17] R. Brown and C. Hanlon. Derivational complexity and order of acquisition in child speech. In Hayes, editor, *Cognition and the Development of Language*. New York: Wiley, 1970.

[18] Greg N. Carlson. Thematic roles and their role in semantic interpretation. *Linguisitics*, 22:259–279, 1984.

[19] Noam Chomsky. *Aspects of the Theory of Syntax*. MIT Press, 1965.

[20] Noam Chomsky. *Language and Mind*. Harcourt Brace Jovanovich, 1972.

[21] Noam Chomsky. *Reflections on Language*. Temple, 1976.

[22] Noam Chomsky. *Lectures on Government and Binding*. Foris, 1981.

[23] Noam Chomsky. *Language and Problems of Knowledge: The Managua Lectures*. MIT Press, 1988.

[24] Peter W. Culicover. *Principles and Parameters: An Introduction to Syntactic Theory*. Oxford University Press, 1997.

[25] S. R. Curtiss. *Genie: A Psycholinguistic Study of a Modern Day 'Wild Child'*. New York: Academic Press, 1977.

[26] Richard Dawkins. *The Blind Watchmaker: Why the evidence of evolution reveals a universe without design*. Norton, 1986.

[27] Terrence Deacon. *The Symbolic Species: The Co-evolution of Language and the Human Brain*. Penguin Books, 1997.

[28] David Dowty. On the semantic content of the notion 'thematic role'. In Gennaro Chierchia, Barbara Partee, and Ray Turner, editors, *Property Theory, Type Theory and Natural Language Semantics*. Dordrecht: Reidel, 1986.

[29] David Dowty. Thematic proto-roles and argument selection. *Language*, 67:3:547–619, 1991.

[30] Robin Dunbar. *Grooming, Gossip and the Evolution of Language.* Faber, London, 1996.

[31] Jeffrey Elman. Learning and development in neural networks: The importance of starting small. *Cognition*, 48:71–99, 1993.

[32] S. M. Ervin. Imitation and structural change in children's language. In E. H. Lenneberg, editor, *New Directions in the Study of Language.* MIT Press, 1966.

[33] Jerry Fodor. *The Language of Thought.* The Harvester Press Limited, 1976.

[34] Edward Gibson and Kenneth Wexler. Triggers. *Linguistic Inquiry*, 25:3:407–454, 1994.

[35] E. Mark Gold. Language identification in the limit. *Information and Control*, 10(5):447–474, 1967.

[36] Stephen J. Gould. Problems of perfection, or how can a clam mount a fish on its rear end. In Stephen J. Gould, editor, *Ever since Darwin: Reflections on natural history.* Norton, 1977.

[37] Stephen Jay Gould and Richard C. Lewontin. The spandrels of san marco and the panglossian paradigm: A critique of the adaptationist programme. In *Proceedings of the Royal Society of London, Series B, Vol. 205, No. 1161*, 1979.

[38] J. H. Greenberg. Some universals of grammar with particular reference to the order of meaningful elements. In J. H. Greenberg, editor, *Universals of Language.* MIT Press, 1963.

[39] Patricia Marks Greenfield and Sue Savage-Rumbaugh. Grammatical combination in *Pan paniscus*: Processes of learning and invention in the evolution and development of language. In Sue Taylor Parker and Kathleen Rita Gibson, editors, *'Language' and Intelligence in Monkeys and Apes*. Cambridge University Pres, 1990.

[40] Arthur C. Guyton. *Textbook of Medical Physiology*. Saunders, eighth edition, 1991.

[41] B Jacennik and M. S. Dryer. Verb-subject order in polish. In D. L. Payne, editor, *Pragmatics of Word Order Flexibility*. John Benjamins: Amsterdam, 1992.

[42] Ray Jackendoff. *Foundations of Language: Brain, Meaning, Grammar, Evolution*. Oxford University Press, 2002.

[43] Simon Kirby. Language evolution without natural selection: From vocabulary to syntax in a population of learners. Technical report, Language Evolution and Computation Research Unit, University of Edinburgh, 1998.

[44] Simon Kirby. Learning, bottlenecks and the evolution of recursive syntax. In E Briscoe, editor, *Linguisitic Evolution through Language Acquisition: Formal and Computational Models*. Cambridge University Press, 1999.

[45] Simon Kirby. Syntax out of learning: the cultural evolution of structured communication in a population of induction algorithms. In D Floreano, J. D. Nicoud, and F Mondada, editors, *Advances in Artificial Life: Proceedings of the 5th European Conference on Artificial Life*. Springer, 1999.

[46] Simon Kirby. Syntax without natural selection: How compositionality emerges from vocabulary in a population of learners. In Chris Knight,

Michael Studdert-Kennedy, and James Hurford, editors, *The Evolutionary Emergence of Language: Social Function and the Origins of Linguistic Form.* Cambridge University Press, 2000.

[47] Simon Kirby. Spontaneous evolution of linguistic structure: An iterated learning model of the emergence of regularity and irregularity. *IEEE Transactions of Evolutionary Computation*, 5(2):102–110, 2001.

[48] Simon Kirby and James Hurford. Learning, culture and evolution in the origin of linguistic constraints. In *Fourth European Conference on Artificial Life.* MIT Press, 1997.

[49] Simon Kirby and James Hurford. The emergence of linguistic structure: An overview of the iterated learning model. In Domenico Parisi and Angelo Cangelosi, editors, *Simulating the Evolution of Language.* Springer, 2002.

[50] Wolfgang Klein and Clive Perdue. The basic variety, or: Couldn't language be much simpler? *Second Language Research*, 13:301–347, 1997.

[51] A. R. Lecours and Y. Joanette. Linguistic and other aspects of paroxysmal aphasia. *Brain and Language*, 10:1–23, 1980.

[52] Gary Lupyan. Modeling syntactic devices: An exploration of language evolution from connectionist and memetic perspectives. Master's thesis, Cornell University College of Arts and Sciences, 2002.

[53] Gary Lupyan and Morten H. Christiansen. Case, word order, and language learnability: Insights from connectionist modeling. In *Proceedings of the 24th Annual Conference of the Cognitive Science Society.* Lawrence Erlbaum Associates, 2002.

[54] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W. H. Freeman, 1982.

[55] Joanna Moy and Suresh Manandhar. Modelling the emergence of case. In *Language Evolution and Computation Workshop, ESSLLI*, pages 42–51, Vienna, August 2003.

[56] Joanna Moy and Suresh Manandhar. In search of inflection. Presented at The Evolution of Language, Fifth International Conference, April 2004.

[57] William O'Grady. *Syntactic Development: the Acquisition of English.* University of Chicago Press, 1997.

[58] William O'Grady. The radical middle: Nativism without universal grammar. In C. Doughty and M. Long, editors, *The Handbook of Second Language Acquisition.* Blackwell, 2003.

[59] Ann M. Peters. *Units of Language Acquisition.* Cambridge University Press, 1983.

[60] Stephen Pinker. A theory of the acquisition of lexical interpretive grammars. In J. Bresnen, editor, *The Mental Represention of Grammatical Relations.* MIT Press, 1982.

[61] Stephen Pinker. *Language Learnability and Language Development.* Harvard University Press, 1983.

[62] Steven Pinker. *The Language Instinct.* Penguin, 1994.

[63] Steven Pinker and Paul Bloom. Natural language and natural selection. *Behavioural and Brain Sciences*, 13:707–784, 1990.

[64] Geoffrey K. Pullum and Barbara C. Scholz. Empirical assessment of stimulus poverty arguments. *The Linguistic Review*, 19:9–50, 2002.

[65] Malka Rappaport and Beth Levin. What to do with θ roles. In Wendy Wilkins, editor, *Syntax and Semantics 21: Thematic Relations.* San Diego, Academic Press, 1988.

[66] Sue Savage-Rumbaugh and Roger Lewin. *Kanzi: The Ape at the Brink of the Human Mind.* John Wiley and Sons, 1994.

[67] Sue Savage-Rumbaugh, Stuart Shanker, and Talbot Taylor. *Apes, Language and the Human Mind.* Oxford University Press, 1998.

[68] Dan I. Slobin. Language change in childhood and history. In J. Macnamara, editor, *Language Learning and Thought.* London: Academic Press, 1977.

[69] Dan I. Slobin and Thomas G. Bever. Children use canonical sentence schemas: A crosslinguistic study of word order and inflections. *Cognition*, 12:229–265, 1982.

[70] Kenneth Smith. *The Transmission of Language: models of biological and cultural evolution.* PhD thesis, University of Edinburgh, 2003.

[71] Kenny Smith and James R. Hurford. Language evolution in populations: extending the iterated learning model. In W. Banzhaf T. Christaller, J. Ziegler, P. Dittrich, and J. T. Kim, editors, *Advances in Artificial Life: Proceedings of the 7th Eurpopean Conferenceon Artificial Life*, 2003.

[72] Luc Steels. Emergent adaptive lexicons. In *Proceedings of the Simulation of Adaptive Behaviour Conference*, 1996.

[73] Luc Steels. Perceptually grounded meaning creation. In *Proceedings of ICAMS, Kyoto*, 1996.

[74] Luc Steels. A self-organizing spatial vocabulary. *Artificial Life Journal*, 2(3), 1996.

[75] Luc Steels. Constructing and sharing perceptual distinctions. In *Proceedings of the European Conference on Machine Learning*, 1997.

[76] Luc Steels. Synthesising the origins of language and meaning using coevolution, self-organisation and level formation. In James Hurford, Chris

Knight, and Michael Studdert-Kennedy, editors, *Evolution of Human Language*. Edinburgh University Press, 1997.

[77] Luc Steels. Structural coupling of cognitive memories through adaptive language games. In *From Animals to Animats 5: Proceedings of the 5th International Conference on the Simulation of Adaptive Behaviour*, 1998.

[78] Luc Steels. Language as a complex adaptive system. In M. Schoenauer et al, editor, *Lecture Notes in Computer Science: Parallel Problem Solving from Nature – PPSN-VI*. Springer-Verlag Berlin and Heidelberg GmbH & Co., 2000.

[79] Luc Steels. Simulating the evolution of a grammar for case. Presented at the 4th International Conference on the Evolution of Language, 2002.

[80] Andreas Stolcke. *Bayesian Learning of Probabilistic Language Models*. PhD thesis, University of California at Berkeley, 1994.

[81] Herbert Terrace. *Nim: A Chimpanzee Who Learned Sign Language*. New York: Knopf, 1979.

[82] Bradley Tonkes and Janet Wiles. Methodological issues in simulating the emergence of language. In Alison Wray, editor, *The Transition to Language*. Oxford University Press, 2002.

[83] Menno van Zaanen. Alignment-based learning versus data-oriented parsing. In Rens Bod, Khalil Sima'an, and Remko Scha, editors, *Data Oriented Parsing*, pages 385–403. Center for Study of Language and Information (CSLI) Publications, Stanford, 2003.

[84] Paul Vogt. Thsim v3.2: The talking heads simulation tool. In W. Banzhaf, T. Christaller, P. Dittrich, J. T. Kim, and J. Ziegler, editors, *Advances in Artificial Life - Proceedings of the 7th European Conference on Artificial Life (ECAL)*. Springer Verlag, 2003.

[85] Paul Vogt. The emergence of compositional structures in perceptually grounded lanugage games. *Artificial Intelligence*, 167(1-2):206–242, 2005.

[86] Paul Vogt. On the acquisition and evolution of compositional languages: Sparse input and the productive creativity of children. *Adaptive Behaviour*, 13(4):325–346, 2005.

[87] B Wilson and A M Peters. What are you cookin' on a hot? *Language*, 64:249–73, 1988.

[88] Alison Wray. Protolanguage as a holistic system for social interaction. *Language and Communication*, 18:47–67, 1998.

[89] Alison Wray. Holistic utterances in protolanguage: the link from primates to humans. In Chris Knight, Michael Studdert-Kennedy, and James Hurford, editors, *The Evolutionary Emergence of Language: Social Function and the Origins of Linguistic Form*. Cambridge University Press, 2000.

[90] Alison Wray. *Formulaic Language and the Lexicon*. Cambridge University Press, 2002.

[91] Alison Wray and Michael R. Perkins. The functions of formulaic language: an integrated model. *Language and Communication*, 20:1–28, 2000.

[92] J. L. Yamada. *Laura: A Case for the Modularity of Language*. MIT Press, 1990.

[93] Willem Zuidema. How the poverty of the stimulus solves the poverty of the stimulus. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems - Proceedings of NIPS02*, volume 15, pages 51–58. MIT Press, Cambridge, MA, 2003.