Computer modeling as a tool for understanding language evolution

Bart de Boer

AI Department, Rijksuniversiteit Groningen

www.ai.rug.nl/~bart

Bart de Boer
AI Department
Rijksuniversiteit Groningen
Grote Kruisstraat 2/1
9712 TS Groningen
the Netherlands

This paper describes the uses of computer models in studying the evolution of language. Language is a complex dynamic system that can be studied at the level of the individual and at the level of the population. Much of the dynamics of language evolution and language change occur because of the interaction of these two levels. It is argued that this interaction is too complicated to study with pen-and-paper analysis alone and that computer models therefore provide a useful tool for understanding language evolution. Different techniques are presented: direct optimization, genetic algorithms and agent-based models. Of each of these techniques, an example is briefly presented. Also, the importance of correctly measuring and presenting the results of computer simulations is stressed.

Keywords: Language evolution, computer modeling, artificial life

1. Introduction

People are fascinated by language, and love to talk and speculate about it. Whenever speakers of different languages or dialects get together, one of the favorite topics of conversation is the comparison between their different languages. Sooner or later, the origins of the differences and possibly the origins of language itself will be discussed. Scientists, not different from other people, like to speculate on the origins of language as well, and the field of language evolution has seen a renewed interest over recent years.

Different questions about the evolution of language can be investigated. When did language evolve? Which of our ancestors had language? Was it a relatively late invention, perhaps as late as 50 000 years ago when Homo sapiens apparently first started to make artistic and symbolic artifacts? Or was it much earlier and did Homo erectus, or perhaps even the Australopithecines already have language? A related question is how fast language has evolved. And *how* language did language evolve? Which evolutionary pressures played a role, and what factors determined that humans ended up with language, while other animals did not? Apart from historical events and circumstances, there are more general processes that determined the evolution of language. These can also be investigated. How much of language evolution is the result of purely biological evolution, and how much of it is cultural? What other factors, besides biological evolution of individual humans can have played a role? What was the role of co-evolution between language and the brain? And that of co-evolution between infants' learning abilities and parenting behavior? What is the role of self-organization, a process often encountered in complex dynamic systems. All these questions have indeed been investigated by different researchers (see e.g. Hurford, Studdert-Kennedy and Knight, 1998; Knight, Studdert-Kennedy and Hurford, 2000; Wray, 2002).

Apart from being an undoubtedly interesting topic, language evolution is also a hard topic to investigate. Language is a complex phenomenon, and evolution is a complex phenomenon, so their relation is by necessity also very complicated. The evolution of human language is also in part the evolution of the human brain. Again, this is a very complex organ, and investigating its evolution is correspondingly complex. Then there is the interaction between human culture, its evolution and the evolution of language. Finally, evolution is a historical process. This means that it has been influenced by coincidences of human history and environment. Unfortunately, our knowledge of the history of human evolution is far from complete, and language itself does not leave any direct historical records, except in the case of written language. However, written language only goes back an insignificant amount of time when compared to the time over which language must have evolved. Therefore, physical evidence of the evolution of language is missing. Only some fossil hints about adaptation for speech exist (e.g. Kay, Cartmill and Balow, 1998; MacLarnon and Hewitt, 1999).

A problem that is both fascinating and hard invites speculation. And indeed, there has been no shortage of speculation about the origins of language. Unfortunately, most of this speculation has been wholly unscientific. For this reason, the Société de Linguistique de Paris in 1866 explicitly forbade all speculation on the origins of language. Still, Jespersen published a book on (among other things) the origins of language in 1922 (Jespersen, 1922). In it, he found it necessary to debunk a number of then current theories on the origins of language. However, the alternative he proposed, human language as derived from song, was not founded any better scientifically. The problem with much of

this speculation was and is that it is trying to find one simple factor that caused language to emerge in humans. However, as has been argued above, the process of language evolution is both complex and dependent on historical coincidences.

Can we do better today? Although the nature of the question has not changed, our knowledge pertinent to the evolution of language has increased enormously. In 1866, the idea of evolution was still very recent: Darwin had only published *On the Origin of Species* (Darwin, 1859) seven years previously. The reality of evolution was still being hotly debated. At the same time, most of linguistics consisted of shoehorning grammars of exotic languages into the grammar of Latin. As for fossil or other physical evidence of language, archeology and paleontology were only just getting off the ground. The first Neanderthal finds had only been made public in 1858 (Schaaffhausen, 1858). Since then, enormous progress has been made in all fields relevant to the study of the evolution of language. Evolutionary theory has developed spectacularly since 1859. We now know about selection pressures, sexual selection, group selection, cultural evolution and many other factors. Because of advances in biochemistry, we also know about the molecular basis of heredity. Our increased knowledge of biology, and of the related field of ethology (the study of animal behavior) has also helped to advance the understanding of the evolution of language. We now know about communication systems in other animals, and about the cognitive abilities of our nearest evolutionary relatives, the great apes. These advances in biology and ethology have gone hand in hand with advances in the understanding of the neural mechanisms that underlie behavior. We have learnt which parts of the brain are responsible for language and cognition and which parts in apes' brains are analogous to these. This has made it possible to form hypotheses about the evolution of the brain. These hypotheses, as well as hypotheses about the evolution of the general anatomy and behavior of humans can be tested objectively because of the paleontological and archeological finds that have been made over the last century and a half. Although fossil evidence will never be abundant, we now have a much more accurate picture of human ancestors. Last but not least, our understanding of what language is and how it works has improved considerably since 1866. Much more is known about what the possibilities of human language are. Many more languages have been described, and these descriptions are nowadays made without reference to the grammar of Latin. Special cases of language have also been described. Pidgin and Creole languages, especially, have shed light on the way new languages can be formed by populations of speakers. With our more extensive knowledge of the possibilities of human language, we can make better theories about the specific human adaptations for language. This growth of our knowledge has made speculation about the origins and the evolution of language more informed and more scientific.

But it is not just the increase of background knowledge that has made it possible to make and test more scientific hypotheses of the origins and the evolution of language. In this paper it will be argued that the use of computer models also constitutes an advance in methodology. Computer models allow researchers to investigate the implications of more complicated hypotheses than would be possible with pen and paper alone. The background of computer modeling in the study of language evolution will be expanded in the next section. Section 3 and 4 are more technical. In section 3 basic techniques for building computer models of language evolution are presented, while in chapter 4 these techniques are illustrated with a few examples. These sections are intended for readers who are interested in building their own simulations, and can be browsed through rapidly by less

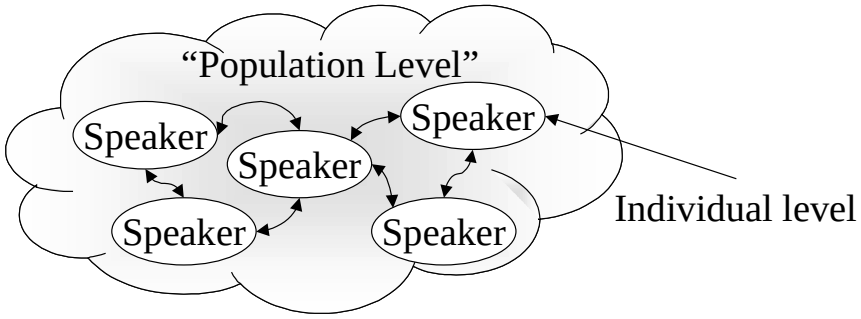

Figure 1: Language can be considered at both the individual level (the knowledge and performance of an individual) and the population level (the linguistic conventions in a population). There are feedback loops between individuals' language and language conventions of the population, making the whole a complex dynamic system.

technical readers. Section 5 discusses the implications of the techniques presented in this paper, and presents some of the conclusions on language evolution that have been reached by computer modelers.

2. The Use of Computer Modeling

In order to understand the use of computer modeling in the study of the evolution of language, we need to understand that there are two levels to language: the level of the individual and the level of the population. These two levels interact and this is an important factor in what makes the dynamics of language in a population so complicated.

At the individual level, language is made up of individual speakers' knowledge of the language, of their limitations in production, of the speech errors they make, of the way in which they acquire language etcetera. This is the level that is related to what Chomsky has called *performance* (Chomsky, 1965) and what De Saussure has called *parole* (De Saussure, 1987). It is studied by psycholinguists who study such things as reaction times in retrieving words and limitations on short-term memory, by researchers of speech errors and speech pathologies, by researchers using neuro-imaging techniques and by researchers of language acquisition. Language at this level is intricately related to the functioning of an individual brain. Also, the language produced by each individual is slightly different.

On the level of the population, language is a conventionalized communication system, with a vocabulary and a set of grammatical rules. The knowledge in the population is uniform to such an extent that users of the language can communicate meanings and intentions with it. This is the level that is related to what Chomsky has called *competence* and what De Saussure has called *langue*. It is often assumed that the language at the level of the population is uniform over space and time. It is also often considered as an abstract system that exists in a sense separately from the individual speakers. Language at the population level is studied in historical linguistics and in general linguistics and is also what is described and prescribed by language teachers.

Both perspectives are equally valid when studying language. It would be impossible to reconstruct the history of a language if one had to take into account the behavior of every individual. It would also be impossible to study organization of language in the brain without looking at the behavior of individuals. However, it is obvious that these two levels do not and cannot exist separately. This is illustrated in figure 1. The population level is an

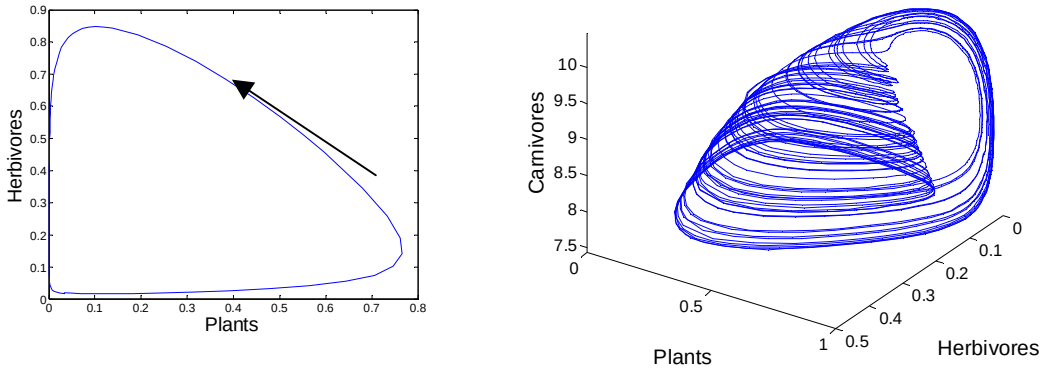

Figure 2: Example of predator-prey dynamics in an ecological model. The equations describing the system are Lotka-Volterra equations, and have been adapted from (Hastings and Powell, 1991). The left plot shows the dynamics of an ecosystem containing plants and herbivores. Such an ecosystem shows cyclic behavior. The right plot shows the dynamics of the ecosystem if a carnivorous predator species is added. Suddenly and surprisingly, the dynamics become chaotic. Such phenomena would be almost impossible to investigate without computer models.

abstraction of the collective behavior of a group of individuals. Behavior on the individual level is influenced by what individuals perceive of the language used in the population of which they are part. The interaction between these two levels is a feedback loop. Changes in behavior of an individual can change the collective behavior and this in turn can influence the behavior of individuals.

These feedback loops are by no means simple. The way language is learnt and the way innovations spread through a population are complex processes. Such systems cannot be described in a mathematically simple way. In a technical mathematical sense they are non-linear systems. As has already been observed by Steels (1998) language is a complex (non-linear) dynamic system. The behavior of such systems is not easy to predict or even to describe. If one makes hypotheses about such systems, they will be extremely hard to test using pen and paper alone. An example of complex behavior is illustrated in figure 2. In this figure the behavior of two ecosystems is compared. When the system goes from two species to three species, the dynamics become surprisingly more complex.

This is where computer models come to the rescue. Once described in sufficient detail, complex dynamic systems can be implemented as computer models. Computers can then simulate the behavior of these models, and provide insights in how they work. When one compares the behavior of the computer model with behavior of the real system, one can check whether the predictions of the theory correspond to what is found in reality or not. Without computer models it would be extremely hard even to check what the exact predictions of the theory are. A common misunderstanding about computer models is that they only produce what has been put in beforehand, and that they are therefore unable to produce any really surprising results. A complex dynamic system's behavior is so difficult to predict that the results of simulating it are often very surprising.

Another advantage of using computer models is that one can use them to do what-if experiments. When studying language evolution or other large and difficult to control

problems, it is often impossible to do controlled experiments. It is possible to observe the behavior of the system under study, but it is not possible to change the initial conditions and see what happens or to restart the system to see what has happened in an earlier phase. Sometimes natural experiments happen, such as when a pidgin or Creole language is formed, but there are always many factors that one does not control. With a computer model, however, one has complete control over all parameters and even over the exact dynamics. One can also run and rerun the model as often as one wants. Computer models therefore make it possible to do as many hypothetical experiments as one wants.

In many fields of science, computer models are indispensable tools for investigating natural systems. One such field is meteorology, and more specifically climate modeling. The earth's atmosphere and its oceans also form a complex dynamical system that would be impossible to understand without computer models. Computer modeling allows us to investigate the long-term dynamics of this system and to perform hypothetical experiments on it by changing parameters and investigating how they influence the model's behavior.

Using computer models to investigate aspects of complex biological systems has since 1989 been the domain of the field of artificial life (Langton, 1989). In this field, mainly biological models are tested using computer simulations. These models can be about behavior of ecosystems (as in the example above) but also about the growth of plants (Prusinkiewicz and LindenMayer, 1990) or on such things as flocking in birds (Reynolds, 1987) or the emergence of ant trails (Colorni, Dorigo and Maniezzo, 1991). From the beginning, artificial life researchers have been interested in using computer models to understand communication, but it wasn't until 1996 when the first conference on the evolution of language was held in Edinburgh, that the application of computer models to the evolution of language got a real boost. Since then the number of papers on computer modeling of language evolution has increased enormously.

Understanding how computer models are made and understanding how to interpret the results from computer models requires understanding of how an abstract system, such as a computer model, and reality map onto each other. Because computer power is limited, and because our understanding of language is limited as well, building a computer model requires us to make abstractions and simplifications. This is not a problem. Simplifications and abstractions are necessary for any scientific theory. Finding the right simplifications is also the key to making successful models of other complex phenomena, such as the example of the climate as mentioned above. However, we should remain aware of the kind of simplifications we make. It is very important not too simplify a model too much, and thus to remove all interesting dynamics. This sometimes happens in systems that are designed for mathematical analysis. Mathematical analysis can only be done on the simplest possible models, and the kinds of models we are interested in are generally not solvable analytically.

Another possible pitfall is to compensate for necessary simplifications in one part of the model by making another part of the model more complicated. Often this only serves to obfuscate the behavior of the system. It is important when modeling a particular problem, to analyze where the simplification bottlenecks of the model are, and not add unneeded complexity elsewhere. We can then build a model that is as simple as necessary and avoids complexity that does not contribute to the realism of the model. Investigation of speech can serve as an example: building a computer model with a very realistic speech synthesizer is not useful if it does not have a correspondingly realistic model of perception. In building

and describing a computer model, it is very important to make our assumptions and abstractions explicit.

When interpreting and presenting results from computer models, we should be aware of how the results of the computer model map onto the linguistic phenomenon under study. For a model of speech sounds this mapping is usually quite straightforward. Such models generally work with direct representations of physical properties of the speech sounds under study. For models of more abstract properties of language, this mapping can be quite intricate. Semantics (meaning) can serve as an example. Meanings in computer models are often implemented as simple numbers that are a measure of how strong the association between a word and an object in the world is. This is easy to implement, but a rather strong simplification of the complexities of semantics in human language. Such more abstract representations require an effort from the author to present the results of the model and from the reader to interpret them. It is therefore essential to clearly communicate the mapping between objects in the computer model and real linguistic entities and to explain how the results of a computer model shed light onto the real linguistic phenomena.

One should also be very careful not to use computer models to investigate aspects of language that they have not been designed for. For example, one can build a computer model for investigating certain properties of speech sounds that does not have a realistic language acquisition component. It would then be disastrous to use this model for investigating language acquisition. Although this is a very obvious example, assumptions and abstractions in a computer model can be extremely subtle. It is easy to forget the exact nature of these assumptions, and the problem gets worse when a computer model that one researcher has designed is used by other researchers.

Deciding which abstractions and simplifications to use is one step in making a computer model. Another step is which computational techniques to use for the computer model. Sometimes the problem one is interested in and the simplifications one has made already determine which techniques can be used. Like the abstractions and simplifications, all different techniques have their advantages and disadvantages.

3. Computer Modeling Techniques

There are many different techniques that are suitable for modeling the evolution of language. Most of these techniques can be divided in three categories: optimization techniques, genetic algorithms and agent-based models. Optimization techniques define a quality measure on (linguistic) systems and try to optimize it. Genetic algorithms are techniques inspired by biological evolution that try to evolve a good linguistic system using a population of candidate solutions. Agent-based models model (a population of) language users as simplified computer programs, and try to emulate how they use language. These categories provide a framework for presenting the different techniques, but it should be kept in mind that they are somewhat arbitrary. There are finer distinctions that can be made within the categories and the boundaries between categories are not always clear.

3.1. Optimization

The hypothesis underlying optimization as a computer modeling technique is that many linguistic structures are in a sense optimized. Different optimization criteria are postulated for different aspects of language. For speech sounds they could be acoustic distinctiveness and articulatory ease. For grammatical constructions, they could be learnability and

parsability. For semantic distinctions and categories, learnability and coverage of the semantic domain could play a role. Cross-linguistic observations and psycholinguistic studies have indeed shown that languages appear to be optimized, at least to a considerable extent. This is relevant in two ways for the study of the evolution of language with computer models. It can be investigated for which factors human language really is optimized, and how the process of optimization is brought about in human language. Optimization criteria can be investigated by generating artificial linguistic systems using different optimization criteria and comparing these systems with real human linguistic systems. If there are important similarities, it is likely that the optimization criterion also plays a role in human language. If the systems are not similar, the criterion probably is not relevant. The second perspective on optimization in language looks at the different ways in which linguistic structures have become optimized. This can have happened through for example biological evolution or cultural evolution. Usually, either genetic algorithms or agent-based models are used for this kind of research, so we will focus on the first kind only in this section.

The basics of implementing optimization are relatively straightforward. A computer model is used to find the linguistic system with the highest quality for some property. Three things are needed for an optimization model. First, a representation of the linguistic system under study is needed. This can be a set of speech sounds, a grammatical system or a vocabulary of words with different meanings. It is crucial that these representations can be modified in small steps in order to optimize them. Second, a quality measure is needed that determines how good a given linguistic system is for the task one wants to investigate. It is advisable to use a function that is easy to calculate and that is smooth. Ease of calculation is important if one wants to make efficient computer models. Smoothness for a quality function is defined as the property that a function gives similar values for similar inputs. In the case of linguistic systems this means that similar systems will have similar quality values. This is important, because most optimization techniques do not work well with quality functions that are not smooth. The third element of an optimizing model is the optimization algorithm. Both the representation of the linguistic system and the quality function depend on the problem one wishes to investigate. Most optimization algorithms, however, are task-independent.

Optimization algorithms generally work by keeping track of the best solution found so far, by making small modifications to this solution and by replacing the old best solution whenever a new solution with higher quality is found. The main differences between different optimization algorithms are in the way new candidate solutions are generated. If very little is known about the behavior of the quality function, the only possibility is often to randomly explore the neighborhood of the best solution found so far. If one knows that the quality function is smooth, one can use a technique called hill-climbing, in which one tries to follow the steepest path up the quality function. A particularly robust optimization technique is simulated annealing (Kirkpatrick, Gelatt and Vecchi, 1983). Here one explores random solutions in the neighborhood of the best solution found so far. Over time, one shrinks the size of the neighborhood in which new solutions are searched. This causes the algorithm to locate the approximate solutions of peaks in the quality function first, and subsequently climb up a promising peak. Both hill-climbing and simulated annealing are illustrated in figure 3.

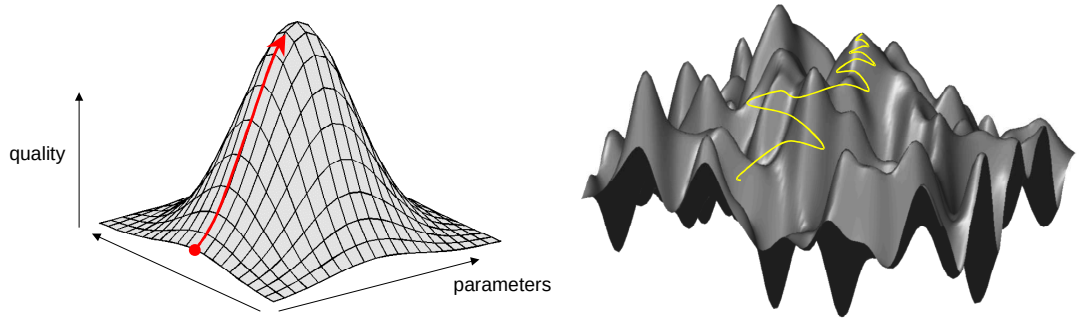

Figure 3: Two different ways of optimizing a quality function. On the left the *hill-climbing* algorithm is shown. It climbs up a peak in the quality function using the steepest ascent. On the right simulated annealing is illustrated. It uses ever decreasing random steps to find the best solution. Note that hill-climbing would most likely get stuck on a sub-optimal peak in this complex landscape.

It is important to note that, except for the simplest possible problems and quality functions, optimization does not always find the best solution. This problem is illustrated in figure 3, where the simulated annealing procedure does not find the highest peak. Straightforward optimization is therefore not recommended when one wants to find the *best* possible solution to a problem. This is however, not usually the case in linguistic problems. Human languages show a fair degree of variation for most if not all properties, even though they are usually close to optimal. An algorithm that manages to find near-optimal solutions most of the time is therefore usually good enough.

Optimization is probably the technique that is least controversial in its applications, as its dynamics are relatively simple: there is an optimization criterion and it results in linguistic systems that are similar to human systems or not. Discussion is possible about the implementation and representation of the linguistic structure, about the quality function that was used, or about the interpretation of the structures that are found, but the optimization process itself is not controversial. The simplicity of optimization is also a disadvantage. It can only be applied to relatively simple problems. As soon as multiple optimization criteria interact, the optimization process becomes more difficult and decisions have to be made about which solutions to investigate. However, optimization is a good technique for checking which criteria play a role in human languages. How these criteria have become important and how the optimization process takes place in human populations must be investigated with different techniques.

3.2. Genetic Algorithms

The second paradigm is that of genetic algorithms (GA's). The genetic algorithm (e.g. Goldberg, 1998) is a technique that is based on the way evolution works in nature. Instead of keeping track of only one potential solution, the algorithm has a *population* of potential solutions. Just as in optimization, it has to be decided how these solutions are represented in the computer model. However, in a genetic algorithm there are two levels of representation: the level at which solutions are evaluated (which is similar to the representation in optimization) and the level in which solutions are recombined and mutated by the genetic algorithm. This is analogous to the distinction in biology between the phenotype (the grown individual) and the genotype (the individual's genes). Analogous to this, solutions in a

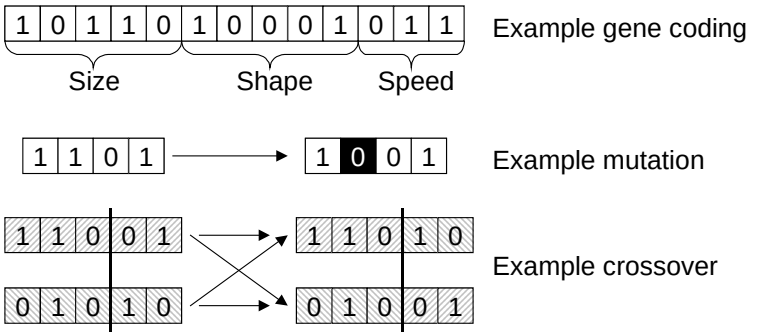

Figure 4: Examples of coding of solutions in terms of genes (for example the size, shape and speed of an animal) and of mutation and crossover operators for generating new genes

genetic algorithm must be representable in terms of artificial genes. In most implementation of genetic algorithms, simple bit strings are used for representing genes. When needed, these genes are converted into possible solutions to the problem at hand (linguistic structures in the case of models of language). These solutions can then be evaluated with a fitness function. This fitness function is comparable to the quality function in optimization. It is a function that gives a high value for good solutions and a low value for bad solutions.

Just as in nature, solutions with a high fitness are allowed to create offspring, while bad solutions are removed from the population. When solutions with high fitness are selected, their genes are used to create new genes for offspring that will replace the bad solutions that have been removed from the population. The idea is that in this way, genes coding for high quality solutions will multiply in the population, while genes coding for bad solutions will disappear.

In order to create offspring based on the parent solutions, combination methods inspired by nature are used. The most important operator is direct copying: most of the time, offspring must be very similar to their parents. Another important operator is mutation. Mutation causes genes in offspring to be different from parent genes. When working with bit strings, mutation generally consists of flipping one of the bits in a gene. Mutation should not be done too often otherwise solutions tend to deteriorate. Another important operator is crossover. In crossover, genes from two parents are combined to form offspring. With this operator one hopes to combine good properties from both parents but it is equally possible that one would combine bad qualities. However, in this case, the resulting low-quality offspring will not be selected for transfer to subsequent generations. Both mutation and crossover are illustrated in figure 4. As in optimization, the right fitness function and the right coding of are essential for the proper functioning of a genetic algorithm.

Many different variants of genetic algorithms exist. There are differences in the exact implementation of the genes and the genetic operators. Often it is important to tailor them to the problem that one wants to solve. Other differences exist in the way one can handle simultaneous optimization of different criteria. The classical GA only optimizes one criterion, expressed in the fitness function. It is of course possible to combine multiple criteria in one fitness function, but as a GA works with a population of solutions, it is also possible to keep all solutions that are the best in each of the criteria one wishes to optimize. This procedure is called pareto-optimization. Making an effort to keep multiple different

solutions is in general a good strategy when using genetic algorithms. Because of the selection process, diversity tends to disappear from the population over time if nothing is done to preserve it. Diversity preserving schemes generally select individuals for procreation by taking into account fitness and how different the individual is from the other individuals with high fitness. If it is very similar, the probability that it will be chosen is diminished. If it is different, the probability will be increased.

GA's are similar to straightforward optimization in that they also optimize on the basis of an optimization criterion (the fitness function), but they are much more flexible and robust. Part of their strength lays in the fact that they keep track of multiple potential solutions. They can therefore be used to model more complex optimization problems and even problems in which the optimization criterion changes over time. Also, the fact that GA's work with a population of solutions makes them more realistic in the case of language. Language is typically used in a group of individuals rather than by a single individual. Finally, genetic algorithms are modeled after Darwinian evolution, and are as such ideally suited for modeling real evolution.

Their resemblance to real biological evolution is possibly the biggest advantage of genetic algorithms when used for research into the evolution of speech. But modelers who enthusiastically embrace genetic algorithms as their paradigm of choice should be aware that there are a large number of design decisions to be made in building a GA for investigating the evolution of speech. Decisions have to be made what to encode as genes and how to implement the fitness function. Another very important point is that one should not confuse biological evolution of the human faculty for speech and cultural evolution of human languages. Historical relations between languages and historical change of languages are often expressed in terms similar to those of biological evolution. It is true that there are definite and valid similarities between the processes of biological evolution and language change, but one should not confuse the two processes in one's model. They are clearly distinct and operate on totally different time scales. They do influence each other, but this influence happens because the properties of a learned system (the language) influence the fitness of individuals that have to learn it. This is an interesting subject of investigation in itself, and is called the Baldwin effect (Baldwin, 1896).

Summarizing, genetic algorithms are a powerful means of optimizing complex systems. This requires selecting an appropriate fitness function and an appropriate representation, both of the linguistic structures that are investigated, as well as their representation as artificial genes. GA's can also be used to study the mechanisms and dynamics of biological and cultural evolution. However, some care must be taken, both by researchers and by readers of papers in which GA's are used, not to confuse these two aspects.

3.3. Agent-based models

Both direct optimization and genetic algorithms assume that there is a property of linguistic systems that can be optimized. This can give interesting results, but it is not always the case that one can identify one easy criterion that is optimized in human linguistic systems. Also, optimization techniques ignore the fact that humans are not optimizers. When humans acquire or use a language, they do not optimize it. They try to conform to the linguistic behavior that they observe. This is an example of the feedback between the individual's use of a language and the use of it in a population. Apparently, over time this results in

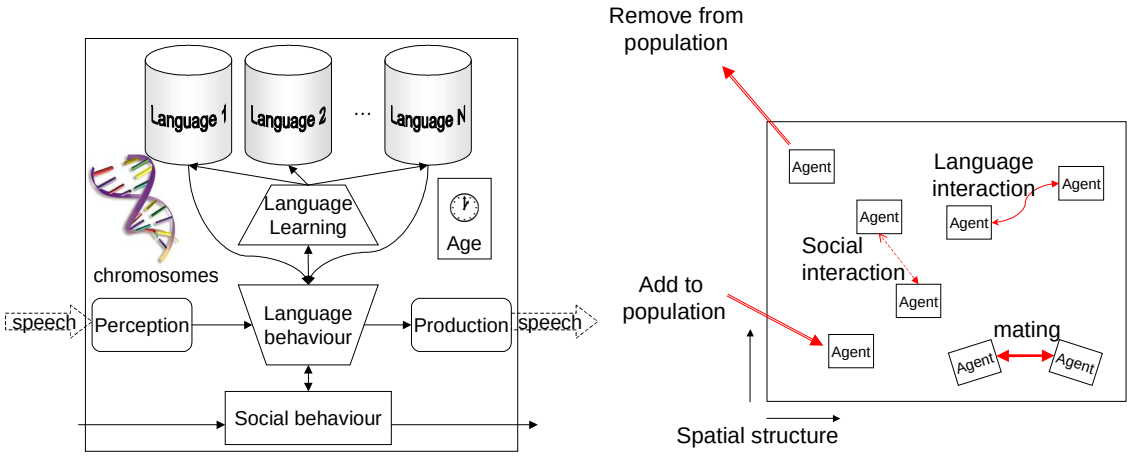

Figure 5: Components of an individual agent (on the left) and possible population dynamics (on the right).

optimization of many properties of language, but how this optimization emerges remains to be explained.

This is where agent-based models find their use. In computer science, agents are small computer programs that can act and interact independently in some limited domain. They are able to perceive aspects of their environment and are able to act on it. This environment is usually simulated, although there are agents that act in non-simulated environments. These are, for example, agents that can act on the internet, agents that can interact with human computer users in a user interface, or even robotic agents that can act in the real world. Often the environment of an agent contains other agents with which it needs to interact. In linguistic agent-based models, individual language users are modeled. These individuals are capable of some limited linguistic feats, depending on what they are used for. Agents that are used for investigating speech sounds are able to perceive, produce and learn speech sounds. Agents that are used for investigating syntax are able to produce, parse and learn syntactically structured utterances. For each linguistic question, specialized agents can be designed. The agents then interact in some way, usually by exchanging linguistic utterances, by observing their (shared) environment and by observing the non-linguistic behavior of other agents. Depending on the interactions, the agents can modify their linguistic knowledge. The influence of the individual actions and interactions on the linguistic systems can then be investigated.

There are more design decisions that need to be taken when constructing an agent-based model than when constructing an optimizing algorithm or a genetic algorithm. Apart from the representation of the linguistic data, decisions have to be made about how the agents interact and how they react to the interactions. On the interaction side, decisions must be taken about what aspects of human interaction must be modeled. Will there be an age structure in the population of agents? Will there be a social structure? Will there be a spatial structure, such that agents that are far apart are less likely to interact than those that are close? Will agents only exchange linguistic information, or will there be non-linguistic interaction as well? Will agents be able to mate with each other and produce offspring? Some of the possible interactions in a population of agents are illustrated in figure 5.

There are also many design decisions that need to be made when constructing individual agents. First of all, it needs to be decided what linguistic utterances these agents can produce and perceive. This is of course determined by what linguistic questions one wants to investigate. These questions also determine what linguistic knowledge must be stored and how it can be learned. It needs to be decided as well whether an agent will be able to learn only one language, or whether it will be able to learn multiple languages. Part of an agent's implementation is determined by the interactions it performs. Furthermore, it should be decided how and if agents change when they become older and how they react to social status. Finally, if an agent-based model is combined with a genetic algorithm, so that agents can produce offspring, it needs to be decided what genes the agent has and how it will mate with other agents. Although agent-based models can become extremely complicated, they are usually kept relatively simple. Making them too complex would result in behavior that is difficult to describe and interpret.

There are two dominant paradigms in agent-based modeling. One paradigm has been introduced by Steels (1995; 1997; 1998) and is called the language-game approach. The other paradigm has been introduced by Hurford and Kirby and is called the iterated learning model (e.g. Kirby, 1999). Both paradigms can be used for investigating any aspect of language. In the language game paradigm, large populations of agents are investigated. These agents are typically egalitarian: there is no distinction between adults and children, or between social classes of agents. It is also typical in this paradigm that agents start out without any linguistic knowledge, and that they "negotiate" a language between themselves. Language games are typically used for investigating cultural or "horizontal" transmission. In the iterated learning paradigm, agents are typically divided in adults and infants. Adult agents produce linguistic utterances, but do not learn, while infant agents learn, but do not produce utterances themselves. At regular intervals, adult agents are removed from the population, infants are turned into adults and new, empty infants are inserted. Populations are also typically small, often as small as one infant and one adult agent. Iterated learning models are typically used for investigating how languages change when they are transmitted from one generation to the next, and which types of language are stable under such "vertical" transmission.

Although much has been said (Kirby, 2002; Steels, 2002) about the differences between the two paradigms, they are really two extremes on the continuum of possible agent-based models. If one considers the space of possible agent based models that vary over both the number of agents in a model and the ratio between the number of horizontal transmissions (within a generation) and the number of vertical transmissions (across the generations) one finds that typical language games are in the corner where there number of agents is large and there are horizontal transmissions exclusively, while typical iterated learning models are in the corner where the number of agents is low and there are vertical transmissions exclusively. Between these two extremes, other agent-based models are entirely possible, and have in fact been investigated.

3.4. Measures and statistics

Computer models, and especially genetic algorithms and agent-based models generate a lot of data. In a simple model for investigating vowel systems (de Boer, 2000) there were 20 agents, each with up to ten vowels that each had three parameters. Such a model requires 600 parameters for its description, and this for every time step. Models that are used for

investigating more complex aspects of language generally have many more parameters. As simulations are run for tens of thousands of time steps, the amount of data generated by a complete run is immense. A human observer cannot interpret such amounts of data. It is therefore necessary to define *measures* on the model that give a reliable indication of its performance. These measures serve as summaries of the model's behavior over time.

It is tempting to choose the optimization criterion used in an optimizing model or the fitness function in a genetic algorithms as the measures. However, this would be wrong. By definition, these values will be optimized. Although the way in which this happens might be interesting in itself (how long does it take, does it continue to change, or does it go to an asymptote etc.) other measures must be monitored in order to learn something about the linguistic aspects of the model.

For clarity of description, it is important that measures are easily understandable by linguists and other non-modelers. At the same time they must give useful information about the way the modeled linguistic system changes over time. Examples of such measures are: average distinctiveness of sounds in a sound system, number of elements in a linguistic system, success of communication between different agents or coherence of the linguistic systems of different agents in a population. Although some measures are more general than others (size of the linguistic system, or coherence in the population are very generally applicable for example) special measures of performance need to be defined for each model.

When these measures have been defined, it becomes necessary to gather statistically significant information about a model's behavior. As in many models randomness plays an important role, this needs to be modeled using the computer's pseudo random number generator. A simulation can then be run many times with different initial values for the random number generator. In this way, a distribution of the different possible outcomes of the model can be generated. Two things must be taken into account in this procedure. First of all, it must be ensured that a proper random generator is used. Some standard random generators are of low quality. Secondly, the kinds of distributions that emerge from simulations of linguistic phenomena are not often normally distributed. One should therefore be careful to apply the correct statistical analysis procedures.

4. Examples

In order to illustrate some of the concepts discussed above, three examples of computer models of language origins will be presented below. Each of these examples illustrates one of the three basic techniques: optimization, genetic algorithms and agent-based models. In order to aid the comparison, all of them model sound systems.

4.1. Optimization: the Liljencrants and Lindblom model

One of the first computer models that was made to investigate factors in the origins of human language was made by Liljencrants and Lindblom (1972). This model was intended to investigate whether the universal tendencies of human vowel systems can be explained as a result of optimization of acoustic distinctiveness. It had been found by linguists that the vowel systems of human languages show a number of regularities: some vowels occur more often than others, and some combinations also occur more often than others. Liljencrants and Lindblom suspected that these regularities could be explained by maximization of acoustic distinctiveness between all the vowels in a language's vowel

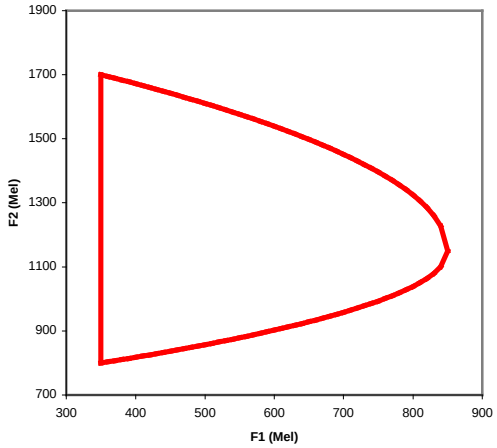

Figure 6: The acoustic space used by Liljencrants and Lindblom (1972). Note that the two dimensions used are the first (F1) and second (F2) formant. Frequencies are in Mel instead of Hertz. Vowels are only allowed within the red area.

repertoire. It was not possible to test this hypothesis analytically, so it was decided to use a computer model.

In this model, vowels were represented as points in an acoustic space. In order for the results to be relevant to linguistic, this space and the representation of the vowels in it had to be perceptually realistic. Phoneticians usually describe the acoustic properties vowels using the first and second (and sometimes third) *formant*. Formants are the resonance frequencies of the vocal tract, and most vowels are distinguished by the lowest two resonances. When represented in a perceptually correct frequency scale (the Mel frequency scale, for example) distances between vowels in the space of the first and the second formant correspond to perceptual distances. Liljencrants and Lindblom therefore decided to represent the vowels in their model by their first and second formants. As humans are not able to articulate every possible combination of two formants, the space in which vowels could occur was restricted to a roughly triangular area. The vowel space is illustrated in figure 6.

In this acoustic space, a variable number of vowels can exist. In order to calculate optimal distinctiveness, Liljencrants and Lindblom consider them as magnets that repel each other. The strength of the force with which they repel each other is inversely proportional with the square of the distance. In this way, the system has potential energy. This potential energy is calculated with the following formula:

$$E = \sum_{i=1}^{N} \sum_{j=1}^{i} \frac{1}{d_{ij}^2}$$

where E is the energy, N is the number of vowels and d_{ij} is the distance between vowels i and j. Both the representation of vowels as points in a two-dimensional space, and the quality of a vowel system as the potential energy in a group of repelling magnets are important simplifications that are however linguistically acceptable.

Just as repelling magnets strive towards a situation with minimal potential energy, vowel systems can be minimized for potential energy. This is done by initializing the vowels to lie on a small circle in the center of the acoustic space. Then each vowel in turn is moved away from this circle in order to decrease the potential energy. The optimization procedure tries to shift the vowel in six different fixed directions over a 100 Mel distance. It keeps the new position that results in the largest decrease in potential energy. When it is no longer possible to move a given vowel away so that potential energy decreases, the next vowel on the circle is tried. This process is repeated until no more reduction in potential energy can be achieved. This particular optimization procedure is an instance of *hill climbing*. For more details, the reader is referred to the original paper.

Liljencrants and Lindblom compare the systems that emerge from their optimization procedure with vowel systems that are found in human language, and find that their model results in realistic vowel systems, especially for smaller numbers of vowels (up to six). The measure they use is the number of vowels that is different between their optimized vowel systems and real human vowel systems, but because of the considerable variation in human vowel systems, this comparison is a bit impressionistic. As the representation of vowel systems in their optimization model is so close to the way linguists represent vowels, the mapping between their results and real vowel systems was straightforward.

4.2. Genetic Algorithms: the Redford model

As has been explained in section 3, optimization models work well when a single criterion needs to be optimized and when the optimization function is relatively smooth. When a problem does not have these properties, a genetic algorithm works better. Syllable systems, as tackled by Redford *et al*. (1998; 2001) have too many complex properties for straightforward optimization, and were therefore modeled with a genetic algorithm.

Redford *et al*. wanted to investigate how properties of human syllable systems can be explained as the result of constraints on perception and production. They also wanted to know what the relevant constraints are. In their model, languages are modeled as collections of words. These words have only form, no meaning. Initially, words consist of random combinations of a small number of phonemes (i, a, u, p, t, k, s, l, n). Phonemes all have a number of binary features, such that distances between them, and therefore between words, can be calculated. The facts that meaning is not modeled and that words are represented as strings of units are important abstractions in this model.

There are a number of perceptual and articulatory pressures on the language. First of all, no two words can be identical. Because of ease of articulation, short words are preferred. Also because of ease of articulation, simple consonant clusters are preferred over complex consonant clusters. Word initial consonants are preferred over word final consonants. Because of acoustic distinctiveness, words should be as different from each other as possible. Finally, because humans produce words by rhythmically opening and closing their jaws (Redford *et al*. call this the mandibular oscillation constraint) adjacent phonemes must differ as much as possible in jaw opening. These different pressures conflict. Preference for short words, for example, is in conflict with maximal distinctiveness. In different runs of their model, Redford *et al*. tested different combinations of constraints to check which ones are needed to produce the most human-like syllable systems.

Words were the units of selection. As words already consisted of strings of discrete units, they did not need to be separately coded as genes. Crossover and mutation were performed directly on the words themselves. In order to assign fitness to words in the population, vocabularies consisting of 25 words were randomly selected from the whole population. For each of these vocabularies fitness was calculated. The fitness of a word was then set to the average of the fitness of all vocabularies in which it occurred. This is a somewhat non-standard way of assigning fitness, but Redford *et al*. use it in order to determine the distinctiveness of words in the language. In a sense, this is a way of preserving diversity, as has been discussed in the description of genetic algorithms. The fittest words were then selected and allowed to create a new language using crossover and mutation.

Redford *et al*. compare the syllable systems that emerge with syllable systems found in human languages and draw the conclusion that the constraints that they have investigated are sufficient to explain human syllable systems, and that it is perhaps not necessary to include both the mandibular oscillation constraint and the constraint against consonant clusters. When operating without the other, both these constraints result in realistic syllable systems. On independent linguistic evidence (infant babbling) they conclude that it is probably the mandibular oscillation constraint that is the one that operates in reality.

4.3. Agent-based models: the de Boer model

The last example that will be discussed is that of an agent-based model that has been investigated by the author himself (de Boer, 1997; de Boer and Vogt, 1999; de Boer, 2000; de Boer, 2001). It is in a sense a continuation of the work by Liljencrants and Lindblom (1972). They provided an explanation of why vowel systems are the way they are: they are optimized for acoustic distinctiveness between the different vowels in the repertoire. However, their model does not provide an explanation of *how* these systems have become optimized. Humans do not explicitly optimize the vowel systems they learn. The hypothesis that was tested with the agent-based model was that the optimization is the result of self-organization under constraints of perception and production in a population of language users.

The model most closely resembles Steels' language game paradigm (Steels, 1997) in that no generations of agents are modeled, but that horizontal interactions (interactions between agents in the same generation) are modeled. It consists of a population of agents that can each produce and perceive vowels in a human-like way. For this purpose they are equipped with a simple vowel synthesizer and a model of perception that, like in the Liljencrants and Lindblom model, is based on formants. Perception of vowels is categorical: an acoustic signal is perceived as the nearest category in an agent's repertoire. Agents are also able to learn new vowels and to modify vowels in their repertoire based on the interactions they have with other agents.

These interactions are so-called imitation games. The goal of the imitation game is to imitate the other agents as well as possible. Imitation was selected as a simplification of real linguistic interactions, as no notion of meaning is required, but the same functional pressures as occur in real language are involved. In an imitation game, two agents that have been randomly selected from the population interact. One agent selects a vowel from its repertoire, and the other agent tries to imitate it, using the vowels in its own repertoire. As the repertoires can be different, it is possible that the imitation this agent produces sounds

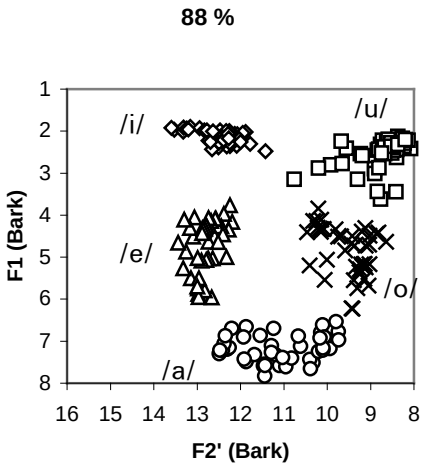

Figure 7: Example of emerged five vowel systems. The vowels are plotted in the space of the first (F1) and second (F2) formants, in the logarithmic Bark frequency scale. Five-vowel systems that emerged from 33 different simulation runs are plotted. It can be observed that the vowel systems for each simulation run are very similar, and that they are very close to the vowel system consisting of /i/, /e/, /a/, /o/ and /u/, the vowel system that occurs most frequently in human languages. This particular five-vowel system occurs in 88% of languages that have five vowels.

quite different from the vowel that was originally produced. If the first agent hears the imitated vowel as the same vowel it originally produced, the imitation game is successful, if not, it is a failure. Depending on the outcome of the imitation game, the participating agents update their repertoire, such that the expected success of subsequent imitation games is increased. In these updates they can only make use of local information: they cannot look into other agents' heads, nor can they do global optimization of their own vowel system. For details on the agents and the imitation games, see (de Boer, 1999; de Boer, 2000; de Boer, 2001).

Agents start out empty, and develop a repertoire of vowels through repeated interactions. As the agents live in a population, part of the challenge is to develop a repertoire that is shared throughout the population. Experiments with the model have shown that shared vowel systems emerge rapidly and reliably in the population. When these systems were compared with human vowel systems, it was found that they are extremely similar. An example of an emerged five-vowel system is shown in figure 7.

Different measures were used to determine the performance of the model. One measure was the Liljencrants and Lindblom energy of the emerged vowel systems. This was shown to be significantly lower than that of randomly generated vowel systems and close to the energy of explicitly optimized vowel systems. This indicates that optimization is an emergent property of the interactions in the population. Another measure was the imitation success of the emerged vowel systems, which was shown to be universally high. This is an indication that the emerged vowel systems were successful for imitation. A final measure was the size of the emerged vowel systems, which tended to be as large as possible

given the amount of noise that was put on the articulations. This too indicates emergence of successful vowel systems. Given that the imitation game uses categorical perception, it would have been trivial to achieve successful imitation with small vowel systems. Finally, vowel systems were compared directly with human vowel systems. This was done in a somewhat impressionistic way by comparing the emerged configurations of vowels with the configurations that are found in human languages, and by checking whether the emerged vowel systems showed the same universal tendencies as those found in human languages. From both comparisons it followed that the emerged vowel systems were realistic.

On the basis of this simulation, a number of variants have been tried. These had mostly to do with adding age structure to the agents, and with removing old and adding new (empty) agents to the population (de Boer and Vogt, 1999). These modifications added vertical transmission to the original model, thus making it resemble the iterated learning model. It was shown that vowel systems could remain stable in changing populations, and that making it harder for old agents to learn increases stability of the vowel systems. This example illustrates that once an agent-based model exists, it is very easy to expand it to do other experiments.

5. Conclusion

Computer models are a useful addition to studying the evolution of language. They can provide insight in the factors that have played and still play a role in making language the way it is, and they can simulate the complex dynamics of language in a population. Neither pen-and-paper analysis nor mathematical analysis can so readily help us to understand such complex phenomena. Many different hypotheses on different aspects of language have been studied successfully with a range of techniques. The examples given above only provide a small taste of what has been done in the case of the sound systems of human language. For overviews of computational models for all aspects of language, see for example (Cangelosi and Parisi, 2002; Kirby, 2002; Christiansen and Kirby, 2003). Most of these models have shown that cultural interactions, functional pressures and general learning mechanisms can explain a lot more about language and language evolution than was previously assumed.

In the preceding sections, it has been shown that there are three different basic techniques for building computer models of human language: optimizing models, genetic algorithms and agent-based models. In some cases these techniques can be combined, for example when agent-based models are combined with a genetic algorithm to model agent evolution. As has been illustrated in the examples, these different techniques can all be applied successfully, depending on what it is exactly one wants to investigate. In building simulations, it also needs to be decided what simplifications and abstractions to make. This is the real art of modeling and useful and meaningful simplifications can make the difference between a usable and an unusable computer model. Finding the right simplifications takes a lot of creativity and effort. Another important aspect of modeling is finding the right measures to describe the performance of a model. Designing the right measures takes creativity as well, but fortunately, measures can often be reused for different simulations.

Of course we should not get carried away by our enthusiasm for computer models. Computer models are just an extra tool in understanding language evolution. We should be careful to combine computer modeling with careful analysis of the available data and with

knowledge and understanding of the available linguistic data. We should take care not to end up investigating the computer model itself, instead of using it for understanding linguistic questions, unless, of course, one is interested in the mathematical aspects of the model. Also, we should be careful not to use a computer model for understanding phenomena that it was not intended to model. This is especially a risk when using computer models that have been developed by others. For example, the agent-based vowel model described above cannot be used for investigating realistic language change, as real language change is influenced by the phonetic context in which sounds occur, as well as the meaning of the words in which they are used. Both aspects are missing in the model.

In any case, it is necessary that we carefully state the assumptions and abstractions that were made when constructing the computer model. The way in which the results of the computer model map back onto real linguistic phenomena also need to be described. This is especially necessary, because many researchers of language evolution are still quite skeptical about the use of computer models. Partly this skepticism is justified, as sometimes too bold claims are made, but a large part of the skepticism is unwarranted and due to a lack of understanding and appreciation of the way computer models work.

There is still a lot that can be done with computer models. A number of open problems remain, even though they have received ample attention from different researchers. The problem of how combinatorial syntax can emerge from non-combinatorial systems has not been understood completely. Neither has the emergence of combinatorial sound systems from holistic utterances. Together these problems would provide insight into the duality of patterning that is so characteristic for human language. Another example of an open problem is the co-evolution of the shape of the vocal tract and the increasing number of distinctions that need to be made for more complex languages. These problems lay at the edge of the understanding of language evolution, and the use of computer models is a promising way of solving these fascinating problems.

References

Baldwin, J. M. 1896. "A new factor in evolution" *The American naturalist* 30:441–451,536–553.

Cangelosi, A. and Parisi, D. (eds) 2002. *Simulating the evolution of language*. Berlin: Springer Verlag.

Chomsky, N. 1965. *Aspects of the theory of syntax*. Cambridge, Mass.: MIT Press.

Christiansen, M. and Kirby, S. (eds) 2003. *Language evolution*. Oxford: Oxford University Press.

Colorni, A.; Dorigo, M. and Maniezzo, V. 1991. Distributed optimization by ant colonies. In *Proceedings of ecal91-european conference on artificial life*, F. Varela and P. Bourgine (eds), 134–142. Paris: Elsevier Publishing.

Darwin, C. 1859. *On the origin of species*: reprinted Penguin Classics, 1985.

de Boer, B. 1997. Generating vowel systems in a population of agents. In *Fourth european conference on artificial life.*, P. Husbands and I. Harvey (eds), 503–510. Cambridge (MA): MIT Press.

de Boer, B. 1999. *Self-organisation in vowel systems*. Brussels: Ph. D. Thesis AI-lab Vrije Universiteit Brussel.

de Boer, B. and Vogt, P. 1999. Emergence of speech sounds in changing populations. In *Advances in artifical life, lecture notes in artificial intelligence 1674*, D. Floreano, J.-D. Nicoud and F. Mondada (eds), 664–673. Berlin: Springer Verlag.
de Boer, B. 2000. "Self organization in vowel systems" *Journal of Phonetics* 28:441–465.
de Boer, B. 2001. *The origins of vowel systems*: Studies in the evolution of language. Oxford: Oxford University Press.
De Saussure, F. 1987. *Cours de linguistique générale, édition préparée par tullio de mauro*. Paris: Payot.
Goldberg, D. E. 1998. *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.
Hastings, A. and Powell, T. 1991. "Chaos in a three-species food chain" *Ecology* 72:896–903.
Hurford, J. R.; Studdert-Kennedy, M. and Knight, C. (eds) 1998. *Approaches to the evolution of languages: Social and cognitive bases*. Cambridge: Cambridge University Press.
Jespersen, O. 1922. *Language, its nature, development and origin*. London: reprinted Allen and Unwin, 1968.
Kay, R. F.; Cartmill, M. and Balow, M. 1998. "The hypoglossal canal and the origin of human vocal behavior" *Proceedings of the National Academy of Sciences* 95:5417–5419.
Kirby, S. 1999. *Function, selection and innateness: The emergence of language universals*. Oxford: Oxford University Press.
Kirby, S. 2002. "Natural language from artificial life" *Artificial Life* 8:185–215.
Kirkpatrick, S.; Gelatt, C. D. and Vecchi, M. P. 1983. "Optimization by simulated annealing" *Science* 220:671–680.
Knight, C.; Studdert-Kennedy, M. and Hurford, J. 2000. *The evolutionary emergence of language: Social function and the origins of linguistic form*. Cambridge: Cambridge University Press.
Langton, C. G. 1989. *Artificial life*. Reading, MA: Addison Wesley.
Liljencrants, L. and Lindblom, B. 1972. "Numerical simulatons of vowel quality systems" *Language* 48:839–862.
MacLarnon, A. and Hewitt, G. P. 1999. "The evolution of human speech: The role of enhanced breathing control" *American Journal of Physical Anthropology* 109:341–343.
Prusinkiewicz, P. and LindenMayer, A. 1990. *The algorithmic beauty of plants*. New York: Springer-Verlag.
Redford, M. A.; Chen, C. C. and Miikkulainen, R. 1998. Modeling the emergence of syllable systems. In *Proceedings of the 20th annual meeting of the cognitive science society*, 882-886. Hilsdale, NJ: Erlbaum.
Redford, M. A.; Chen, C. C. and Miikkulainen, R. 2001. "Constrained emergence of universals and variation in syllable systems" *Language and Speech* 44:27–56.
Reynolds, C. W. 1987. "Flocks, herds, and schools: A distributed behavioral model" *Computer Graphics* 21:25–34.
Schaaffhausen, H. 1858. "Zur kenntniss der ältesten rassenschädel" *Müller's Archiv für Anatomie, Physiologie und wissenschaftliche Medicin*:453–478.
Steels, L. 1995. "A self-organizing spatial vocabulary" *Artificial Life* 2:319–332.
Steels, L. 1997. "The synthetic modelling of language origins" *Evolution of Communication* 1:1–34.

Steels, L. 1998. Synthesising the origins of language and meaning using co-evolution, self-organisation and level formation. In *Approaches to the evolution of language*, J. R. Hurford, S.-K. Michael and C. Knight (eds), 384–404. Cambridge: Cambridge University Press.
Steels, L. 2002. Iterated learning versus language games. Two models for cultural language evolution. Paper presented at *International Workshop of Self-Organization and Evolution of Social Behaviour*, Monte Verità, Ascona, Switzerland.
Wray, A. (ed.) 2002. *The transition to language*. Oxford: Oxford University Press.