*Bulletin of The British Psychological Society* (1987), **40**, 121–129 *Printed in Great Britain*

# Reasoning, imagining, and creating*

**P. N. Johnson-Laird**

Stuart Sutherland (1976) tells the story of a patient at the Maudsley who was suffering from chronic depression. He had been on a regime of antidepresants and he had received a course of ECT treatment . . . to no avail. Then one day he heard that he had inherited a large sum of money. He leapt from his bed, discharged himself from hospital, and was evidently cured. This form of psychotherapy is not listed in my A to Z guide to the 250 different therapies in use today (Herink, 1980). Yet I can corroborate a version of it. In a week when my latest *magnum opus* was rejected by *Psychological Review* and my latest experiment failed (after a long row of failures) I was close to a mid-career crisis. Then I heard that I had received the Presidents' Award. I want to thank the President and the other officers of the British Psychological Society, and of course my colleagues and collaborators, and to assure them that their form of therapy also works.

My theme in this paper is thinking of various sorts, particularly reasoning and imagining. Students of deductive reasoning often argue that people are not very good at it because they are too imaginative: they introduce extraneous premises and fail to stick to the logical task (see e.g. Henle, 1962). Students of creativity, however, often argue that people are not very good at generating new ideas because they are too logical: their thoughts run only along well-worn rational tracks. In the best tradition of British compromise, I want to say: a plague on both your houses. Reasoning *is* an imaginative process.

## Reasoning and imagining

The best way that I can demonstrate the role of imagination in reasoning is to give you a simple reasoning task. Suppose that you are serving on the jury of a murder trial and that two points are established beyond reasonable doubt:

1. The victim was stabbed to death in a cinema during the afternoon showing of *Bambi*.
2. The suspect was on an express train to Edinburgh when the murder occurred.

What conclusion would you draw?

At this point in the lecture, a dialogue occurred between the audience and the lecturer. The gist of it is reported below:

MEMBER OF THE AUDIENCE: The suspect is innocent.

LECTURER: Yes, that is the conclusion initially drawn by the majority of subjects whom Tony Anderson and I have tested. But are you sure? Is there any other possibility?

MEMBER OF THE AUDIENCE: Perhaps there was a cinema on the train.

LECTURER: It's possible, but in fact there are no cinemas on express trains to Edinburgh. Any other possibilities?

MEMBER OF THE AUDIENCE: An accomplice.

LECTURER: It's possible, but in fact the suspect did not have an accomplice. Any other possibilities?

MEMBER OF THE AUDIENCE: Perhaps he left some sort of spring-loaded knife in the seat.

ANOTHER MEMBER OF THE AUDIENCE: Or a knife suspended above the seat in a block of ice.

LECTURER: Yes, he could have done. But as a matter of fact, he didn't. Nor did he use a radio-controlled robot—another of our subjects' suggestions—to stab the victim. Any other possibilities?

The audience remains silent.

LECTURER: Well, there is one other devilish possibility, which I will tell you in a moment to give you a chance to think of it for yourselves.

The initial conclusion that the suspect is innocent is a typical example of everyday reasoning, and it illustrates several important phenomena. For instance, when you reason in this way, you depend both on the premises and on your general knowledge, e.g. you know that one person cannot be in two places at the same time, and that one needs to be near someone in order to stab them. You forge these links in the inferential chain so rapidly and automatically that you are hardly aware of them. There are infinitely many logically valid conclusions that follow from the premises, but most of them are totally trivial, such as a mere conjunction of the premises:

> The victim was stabbed to death in a cinema during the afternoon showing of *Bambi*, AND the suspect was on an express train to Edinburgh when the murder occurred.

You do not draw such a banal, though valid, conclusion. You draw an informative conclusion that is not explicit in the premises. Hence, you must be guided by some principles outside logic. In fact,

---

*Presidents' Award paper presented at the BPS Annual Conference, Sheffield, April 1986.

your conclusion is invalid, and no matter how much information you are given about the circumstances of the murder, you are unlikely ever to be able to infer validly that the suspect is innocent. Inferences in daily life are seldom deductively closed. Yet, although your conclusion is invalid, if it is challenged, then you can test its validity. When Tony Anderson and I expressly queried such conclusions in a series of unpublished experiments, our subjects (like the audience in the lecture) searched for alternatives and often produced scenarios in which the subject is guilty.

These various phenomena imply that people do not ordinarily reason by following the formal rules of inference of some tacit mental logic (*pace* Inhelder & Piaget, 1958; Henle, 1962; Osherson, 1975; Johnson-Laird, 1975; Braine, 1978; Rips, 1983; and many others). On the contrary, they reason by carrying out three main operations (Johnson-Laird, 1983):

1. They imagine the state of affairs described by the premises, i.e. they construct a mental model based on the meaning of the premises and on their general knowledge.
2. They formulate an informative conclusion true in the state of affairs characterized by the model.
3. They search for alternative models of the premises that would be counter-examples to their conclusion.

In short, reasoning is not a formal syntactic process that proceeds in a mechanical way; it is a semantic process that depends on imagining states of affairs and on searching for counter-examples.

How might such a search proceed? It probably depends on several distinct procedures. One method consists in straightforward spatial manipulations. The initial model represents the train and the cinema as spatially separate. They can be moved together in the mind's eye. This manipulation suggests, as subjects are wont to do, either that there might be a cinema on the train, or else that the train rushed through the middle of the cinema, or near to it, and the suspect was able to lean out of the window and stab his victim with a long knife! Another method depends on thinking of the specific event as an instance of a general class, and then using general knowledge about that class to yield a specific method. For example, the murder is an instance of a crime, and part of general knowledge about crimes is that they can be committed by an accomplice.

A variation on this method calls for a species of analogical thinking. If it occurs to you that the murder *could* be an instance of an action at a remote distance, then part of general knowledge about such actions is that they can be carried out by automatic devices. This idea in turn readily leads to the notions of spring-load knives, robots, etc. The difficulty is to think of remote action in the first place. Likewise, if it occurs to you that the victim could have killed himself, then you may be led to a specific way in which the suspect might be guilty. Of the several hundred people to whom I have given the problem, two spontaneously suggested the following ingenious possibility: the suspect gave the victim a post-hypnotic suggestion to stab himself during a certain climactic scene in the film. (Jerry Bruner was one of the people who made this suggestion; the other, according to my informants, was a Swedish princess.)

There are several theories of analogical thinking. They concentrate on the process by which high-level relations, either structural (Gentner, 1983) or semantic (Gick & Holyoak, 1983), can be transferred from one domain to another as an aid to solving a problem. But the critical difficulty, as I have argued elsewhere (Johnson-Laird, in press), is to find the right domain in the first place, that is, to think of action at a distance or of suicide as relevant to the murder in the cinema. There are too many potentially relevant domains for a simple search procedure to succeed. I conclude that the search for counter-examples in everyday reasoning often depends on an exercise of creativity.

## Creativity from a computational standpoint

If reasoning depends on a creative search, one is bound to ask: what is creativity? Psychologists have, of course, made many attempts to answer this question (see Perkins, 1981, for an excellent review). In my opinion, the best way forward depends on an analysis from a computational standpoint. I will begin with a working definition of creativity that has three clauses. First, the process of creativity does not depend merely on recalling some existing idea. When you remember that criminals have accomplices, you are hardly being creative. Second, a creative product is not merely the result of calculation or of some other deterministic mental process. When you multiply two numbers together in your head, then you are hardly being creative—even if you get the wrong answer or the result is a number that you have never thought of before. Third, creation always requires that its products meet some existing criteria or constraints. A creative scenario for the murder in the cinema must meet the constraints of the problem. Likewise, novels and poems, sonatas and symphonies, theories and hypotheses, all have their own constraints. There are definite genres of creativity, and even the creation of a new genre must meet certain constraints—not anything goes.

The only clause in this working definition that may cause trouble is the second one: the notion that creative processes are not deterministic. This concept comes from the theory of computation and I need to explain what is at stake here.

Consider a grammar of English. Somewhere amongst it there will be a set of rules for verb phrases. One rule will allow that a verb phrase can consist of a transitive verb followed by a noun phrase, as in the sentence 'John told a story':

VP → Verb-transitive NP

and another rule will allow that a verb phrase can consist solely of an intransitive verb, as in 'Mary laughed':

VP → Verb-intransitive.

Which rule should be used in *producing* a sentence? The grammar allows either rule and says nothing about how the choice should be made. In this respect, the grammar is *non-deterministic*, to use an ugly but useful piece of jargon.

If you have seen Clouzot's film, *The Picasso Mystery*, you will recall that you watch Picasso painting several pictures. If the film is stopped just before he makes a brush stroke, then he might be about to do a stroke that curves upwards or one that curves downwards. In either case, the result will be a Picasso painting. Hence, whatever it is that constitutes a Picasso painting does not determine the precise stroke that should be made at each point in the making of a picture. From a computational standpoint, the mental processes underlying the production of a Picasso—or indeed any other creative product—are not deterministic.

There are different interpretations that you might wish to place upon this claim. You might say that if we knew the state of Picasso's bank balance, or of his digestion or love life, or the direction in which the wind was blowing, then one day we would be able to predict precisely which brush stroke Picasso would make. In other words, non-determinism is merely a label for our ignorance. Or you might say that human beings have the capacity to make arbitrary choices, and that Picasso merely makes an arbitrary choice amongst various possible brush strokes. Experiments have shown that people are not good at performing in a purely random manner (e.g. Baddeley, 1966), but it does not follow that they have no machinery for making arbitrary choices. Perhaps they can make such choices—even unconsciously—by the mental equivalent of spinning a coin (albeit a procedure in which one trial is not altogether independent from another). Still another interpretation is that mental processes ultimately depend on quantum events, which are truly non-deterministic, and this property is accordingly reflected in Picasso's decision. I do not know which, if any, of these interpretations of non-determinism is correct, though I suspect that human beings do have the ability to make arbitrary choices. Since there is no obvious way to decide amongst them on empirical grounds, I shall be agnostic. Fortunately, in what follows it makes little difference which interpretation is made.

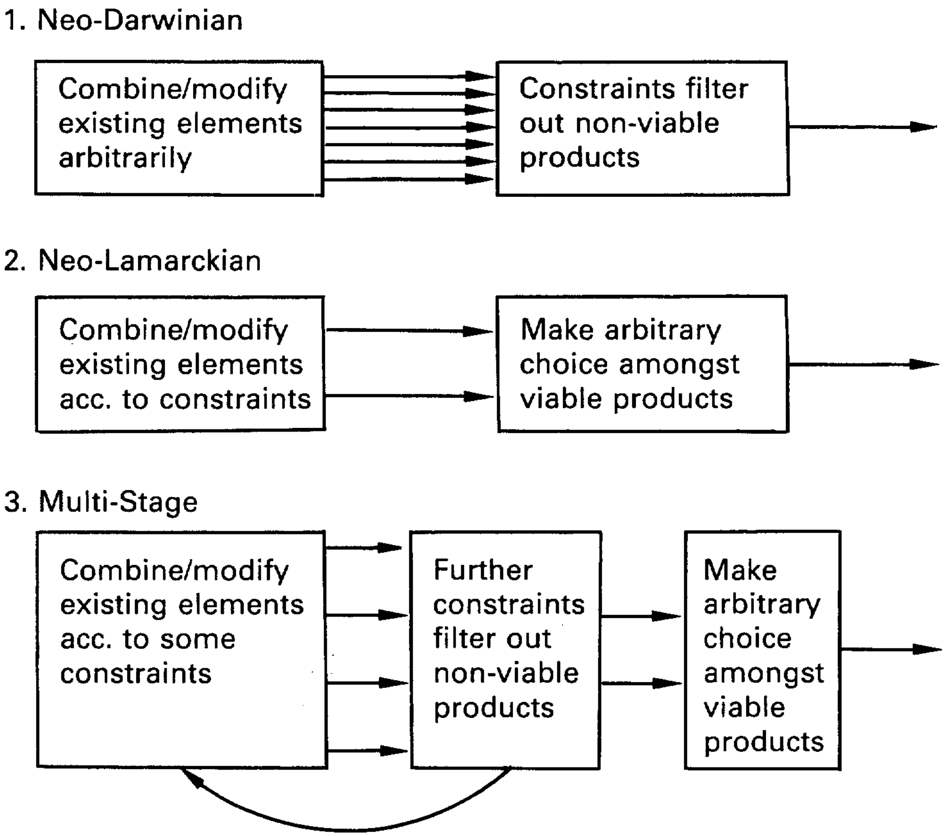

**Figure 1.** Three classes of creative algorithm.

## The architecture of creative algorithms

Any process that does not depend on magic can be modelled by an algorithm, i.e. a finite set of instructions for an automaton such as a computer. This claim, which is sometimes known as Turing's thesis, lies at the heart of the theory of computability. It cannot be proved, but it could be refuted by showing that it is possible to characterize an effective procedure for some task, which could not be executed by an automaton such as a computer. Perhaps human creativity does depend on magic—on processes that cannot be given a scientific explanation. Perhaps it depends on quantum events that are not algorithmic and that would refute Turing's thesis. If we take an optimistic view, however, then a rather striking consequence emerges from our working definition of creativity. There are only three possible sorts of algorithm that could underlie creative mental processes.

A creative process must start with some existing 'building blocks', and our working definition entails that it must be both non-deterministic and meet some criteria. One class of algorithms combines the building blocks in an arbitrary way, and then uses the criteria to filter out the rubbish. The arbitrary generative stage ensures that the process is non-deterministic, but obviously most of its products will be nonsensical. The procedure is analogous to the neo-Darwinian theory of the evolution of species, and I shall use the same label to refer to this sort of creative algorithm. Another and much more efficient class of algorithms uses all the criteria in the initial generative stage. This procedure, which I shall refer to as 'neo-Lamarckian' for obvious reasons, ensures that only viable products are produced. However, since by definition there will often be more than one possible step at a given stage of the generative process, it will be necessary to choose which particular step to take. Since all the criteria are taken into account in generating the range of choices, the selection amongst them will have to be made arbitrarily. The neo-Darwinian algorithms apply the criteria to select amongst the products of the generative stage, and the neo-Lamarckian algorithms apply the criteria *ab initio* in the generative stage. There remains only one other possible class of algorithms: those that use some criteria in generating initial ideas and other criteria in subsequent selective stages. Indeed, there may be a whole series of such stages, or equivalently feedback from a selective to a generative stage. Even when all the criteria have been applied, however, there will still be more than one possible result because the process is non-deterministic. Hence, at various points, arbitrary choices will have to be made. I shall refer to these algorithms as 'multi-stage'. Figure 1 summarizes the three classes of algorithm.

## Musical improvisation: A case history

When you have a set of just three classes of algorithm for carrying out a task, it is an excellent idea to determine which of them might be employed by human beings. Rather than consider reasoning, which is a difficult case because it depends on meaning, I am going to examine a domain that is more approachable from an algorithmic standpoint, namely, music. The profound psychological question about music is why it should be so popular. Music consists of essentially abstract patterns in time, and these patterns usually have no denotation outside themselves. Music ought, therefore, to be about as popular as, say, the paintings of Jackson Pollock, which are also abstract patterns. (Pollock was a fine painter, but not even his most ardent admirers would claim that his work has a popular appeal.) Psychologists often say that music is popular because it stirs the emotions, but this response merely replaces one question with two: how does music move the emotions? and why do people like having their emotions stirred in this way? I confess that music's popularity remains extremely puzzling to me.

The advantage of studying music is that it can be treated as a purely formal, syntactic exercise; and at least one great composer, Stravinsky, argued in his *Poetics of Music* that music should be composed and listened to as purely formal patterns in time. By studying musical improvisation, a psychologist accrues yet another advantage: improvisers perform in real time, that is, they cannot go back and change what they have played. Hence, the principles of musical improvisation must be represented within their minds, and these principles must be sufficient to generate acceptable music if the musician is to continue in gainful employment. So I will begin with the case of musical improvisation.

One feature of most systems of improvisation is that they depend on two quite distinct psychological components: a set of basic structures that are committed to memory, and a set of tacit skills that construct an actual improvisation from a particular basic structure. In the case of Indian classical music, the basic structures consist of scale-like patterns known as 'ragas' around which the musicians weave their improvisations. In the case of modern jazz, the basic structures consist of sequences of chords, often borrowed from popular music, and the musicians improvise novel melodies that fit the particular chord sequence of the piece. The basic structures are readily accessible to consciousness: musicians can describe them in detail, teach them explicitly, and compose new structures. The tacit skill of improvisation, however, is no more accessible to consciousness than our skill in putting words together when we speak spontaneously. The skill is acquired first by imitating what other musicians play, and then by attempting to improvise for oneself. One learns to improvise by improvising—at first disastrously, but with perseverance the basic skill can be mastered. The great improvisers are those that define the genre by setting the standard with their own highly original performances.

A critical datum is the speed with which musicians can improvise a genuinely novel melody. Tempi of 10 to 12 notes per second are not uncommon in the case of modern jazz. I am going to advance a computational conjecture about the mechanism that makes such tempi feasible, but as a preliminary I need to explain the fundamental notion of *computational power*, which plays a central role in my conjecture. One computational device is more powerful than others if it can compute things that they are unable to compute. You might imagine that power therefore depends on having access to a richer set of basic instructions. Surprisingly, although such instructions may enable a computation to be carried out more efficiently, they are not essential for an increase in what can be computed given a certain minimal set of instructions. The root of power is memory.

Consider, for example, the task of mental addition: I give you a digit from each of two numbers, working from right to left, and you reply with the sum of the current two digits before I present you with the next pair from the two numbers. All you have to remember is whether there is a carry or not, since the carry, if there is one, is always 1. An automaton can be programmed to carry out this task in a way that does not require any memory whatsoever for the results of intermediate computations: it just has two sets of rules for adding pairs of digits, one set for where there is a carry, and another set for where there is no carry. Now consider the task of multiplying two numbers. This task is equivalent to summing, not just a pair of integers, but some arbitrary number of integers. No automaton can carry out this task without having recourse to a memory in which to store the results of intermediate computations—just as you would need to store such results if you were trying to multiply two numbers in your head. Thus, multiplication calls for a computational device that has more power than the device that is required for addition.

The output of any automaton regardless of its computational power can always be characterized by a grammar. Just as automata differ in their computational power so too grammars differ in their power. Now I can tell you my conjecture about musical improvisation. It is that the tacit principles used to improvise melodies should have the weakest possible computational power. They should not rely on any memory for the results of intermediate computations, but rather should immediately lead to the generation of a note—just as you generate the sum of the two integers by speaking out loud the sum of each pair of digits. The advantage of producing musical notes in this way is speed: things go faster if you don't have to remember intermediate results. In order to ensure that the results of improvisation are interesting, however, the other half of my conjecture is that the basic structures that are composed 'off line' are constructed with a high degree of computational power.

How can we test this conjecture? Obviously, we have no direct method of examining the principles underlying an improvisation. Our situation is entirely analogous to that of a linguist, who wants to characterize the principles of an unknown language. Our best hope is to examine a corpus of musical improvisations, such as those of modern jazz, with the aim of determining the sort of grammar that is needed to characterize them. We need to construct at least two different grammars: one that underlies the generation of the basic structures—the chord sequences—and one that is used in a transducer that takes a chord sequence as input and then generates an improvised melody that fits it. The prediction that follows from my conjecture is, of course, that the grammar underlying chord sequences will be of a greater computational power than the grammar underlying the tacit skills in the transducer.

My method has indeed been to try to characterize a corpus of improvisations, but with one important caveat: unlike the majority of linguists, I have attempted to test the adequacy of my grammars by embodying them in a computer program that generates music. The advantages of this procedure are twofold: it readily reveals inadequacies in the grammars, which otherwise would be very difficult to detect, and it establishes that the algorithmic claim that I am making is at least coherent and does not take too much for granted. The disadvantage of this procedure is the difficulty of explaining how an algorithm works without swamping you with an overwhelming amount of technical detail. I have implemented three different algorithms, which produce, respectively: improvised bass lines, improvised melodies, and tonal chord sequences of the sort that underlie such improvisations. I will sketch the outlines of each of them.

There are several possible hypotheses about how a jazz bass player makes up a bass line. One possibility is that the player merely chooses as the next note to play any note that is amongst those making up the current chord in the chord sequence. For example, if the current chord contains the notes G, B, D, and F (G dominant seventh), then the player chooses any of these notes. Since the range of a double bass is about two octaves, the player will have on average about 8 notes to choose from on each beat of the bar. That is enough degrees of freedom to allow a vast number of possible bass lines for any given chord sequence. However, it is obvious that bass players do not improvise in this way. On the one hand, they often choose so-called 'passing notes', which are not amongst the notes in the current chord, but which typically lead from one such note to another in a way that enhances the melodic character of the improvisation. On the other hand, bass players do not leap wildly around from a low note to a high note—a manoeuvre that is perfectly within the bounds of the present hypothesis.

Another possibility is suggested by Ulrich's (1977), and more recently Levitt's (1981), general account of jazz improvisation. They suggest that musicians learn by heart a vast repertoire of 'motifs', i.e. melodic fragments, from which they select a series of instances that they tie together to form a melody. While it is true that all improvisers use motifs some of the time, I do not believe that any competent improviser uses them all the time. It is much less of a load on memory (and therefore requires much less computational power) to learn to make up new melodies than to remember a vast number of existing melodic fragments, to select instances of them one after another, and to modify them to fit the current harmonic sequence. This view is confirmed by the intuitions of the performers themselves, by the ethnomethodologist David Sudnow (1978) who recounts how he learnt the art of jazz improvisation, and by the examination of corpora of improvisations (see e.g. Perlman & Greenblatt, 1981). A grammar is a machine for generating motifs, but it will generate ones that the musician has never played before. Some such machine appears to be necessary even assuming the theory of Ulrich and Levitt, since the motifs to be learned must be invented by someone.

My hypothesis is that bass players choose their notes in order to meet both the harmonic constraints of the chord sequence, and a tacit knowledge of what constitutes a pleasing melodic contour. Roughly speaking, players know that after a certain number of small steps in the pitch of the bass line it is about time to take a large step in pitch, and vice versa. Such a variety makes for a pleasing melodic contour. My algorithm is thus a transducer that takes a chord sequence as input, and produces an appropriate—well, largely appropriate—bass line as its output. On each step of the program, it generates a musical interval according to the rules of a *regular* grammar of contours, and then selects a note that meets the constraints of the size of the interval and a few simple harmonic principles, e.g. if the note is on the first beat of the bar and coincides with the start of a new chord, then it should be a note within the chord rather than a passing note. Where more than one note meets all the constraints, then an arbitrary choice is made between them. (In a computer program, arbitrary choices depend on a technique borrowed from the casino at Monte Carlo: a random number is generated, and its magnitude is used to determine which choice to make.)

A fragment from a typical output of the algorithm is shown in musical notation in Fig. 2. Some outputs are slightly better than this example, and some are slightly worse, but it is typical. The output of the program, which also generates a rudimentary accompaniment, is played by way of a further program devised by my colleagues Roy Patterson and Rob Milroy, which synthesizes the sound of a double bass and the accompaniment.
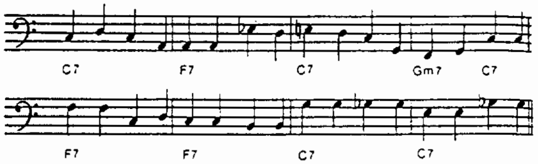
**Figure 2.** A typical fragment from the output of the program generating bass lines.

If I had a group with such a bass player, then I don't think I would sack him or her immediately, but there are some obvious defects in the improvisations. The major solecism reveals the existence of a special category of passing notes. Until I heard the output of the program, I did not realize that so-called 'flattened fifths' had to be treated with more care than other passing notes: the preceding note should tend to be the root or fifth of the chord, or else the sequence may have entirely the wrong harmonic implications (cf. the eighth bar in Fig. 2, which does not sound right). The other defects of the program are that it makes no use of chromatic runs, motifs, or complicated rhythms. It is not my aim, however, to render human bass players obsolescent.

Even if these various defects were corrected, the program would still make use of only a minimal memory. It stores the current item generated by the contour grammar until a note is selected, but, as I have mentioned, this grammar is *regular*, which means that there are no results of intermediate computations to be recorded. The program also has a working memory for the current position in the chord sequence, and a buffer that stores the previous note produced by the program. My conclusion is that viable bass lines can be generated by algorithms that use only a minimum of computational power.

A melody consists of a sequence of notes that each have a specified pitch, duration, loudness, and manner of articulation; it may also contain rests, i.e. silences of a specified duration. My algorithm for melodies ignores loudness and manner of articulation in order to concentrate on pitch and rhythm. Hence, it merely adds a grammar for generating rhythms to the bass program. It takes as input a tonal chord sequence and chooses a series of notes and rests on the basis of two regular grammars: one that generates melodic contours, and one that generates rhythms, i.e. the relative durations of notes and rests. If you tap in synchrony with a tune, then you are tapping out its rhythm; and, since you can recognize tunes 'performed' in this way, the critical durations must be the intervals between the onsets of notes. Jazz improvisations like any other melodies are made up of phrases—much as discourse is made up of separate utterances—and the program produces separate phrases. It uses a regular grammar to generate the next duration in a phrase. If the item is to be a note as opposed to a rest, the contour grammar generates a step in melodic contour, and finally the program selects a note on the basis of this information and harmonic constraints like those of the bass program. The major consequence of this design, which again relies on a minimum of computational power, is that the choice of the current pitch and duration are not governed by any information about the specific notes to be played next. They are constrained, but only by the information that can be built into the regular grammars for rhythm and contour, and into the set of harmonic principles.

The grammars that I have so far used to test the program were derived from Christmas carols. Carols have a simple structure in comparison with modern jazz. Figure 3 presents two representative examples of the output of the program, which to my ear sound typical of Christmas carols. Whether the same minimal degree of computational power will suffice for the melodic improvisations of modern jazz remains an open question, but in the process of developing the required grammars I have yet to discover any phenomena that would force me to abandon the format of regular grammars.
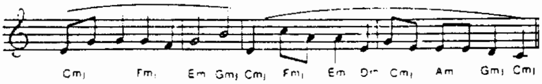
**Figure 3.** Two examples of phrases generated by the program for melodies.

The third algorithm that I have developed produces tonal chord sequences. Tonality is the basis of most of the familiar forms of Western music, and it relies, as Christopher Longuet-Higgins (1979) has shown, on a three-dimensional conceptual space that organizes the relations between pitches. Although there is a large theoretical literature on tonal chord sequences (see, for example, Forte, 1979), when the theories are translated into explicit terms, they invariably turn out to be equivalent to *regular* grammars. But such grammars are not powerful enough to account for the chord sequences of modern jazz, or, I assume, for those of other sorts of tonal music. The basis of this claim is a phenomenon that is familiar to musicians. In many cases, a jazz chord sequence is developed as a variation on an underlying sequence, which is either traditional or else borrowed from popular music. Mark Steedman (1982) has outlined a grammar that generates variations on a given underlying form. His work inspired my algorithm, since it seemed possible that if one devised an algorithm that generated the underlying forms too, then some principles in Steedman's grammar would no longer be necessary. In fact, my algorithm vindicates his analysis.

The algorithm contains three stages. The first stage uses a simple grammar to generate sequences involving the tonic, dominant, and a few other chords. The second stage takes the resulting sequence as input and then uses 'context-sensitive' rules like those in Steedman's grammar to make various interpolations and substitutions. The third stage uses a further set of context-sensitive rules to make some final substitutions of one sort of chord for another.

Jazz musicians recognize that there are many variations on, for example, the underlying sequence:

| I | V |

(non-musicians need not worry about the interpretation of these symbols, and should treat the problem as one in which acceptable strings in an abstract symbolic language have to be generated.) This sequence, which is produced by the first stage of the algorithm, can be transformed by the second stage, which makes interpolations according to the so-called 'cycle of fifths', into the following sequence:

| Imj7 VIm7 | IIm7 V7 |

Alternatively, the second and third stages can produce such variations as:

| Imj7 bIII7 | bVImj7 bII7 |

or:

| Imj7 IVm7 bVII7 | bIIIm7 bVI7 IIm7 V7 |

Of course there are many possible algorithms that could generate the required chord sequences, but what is clear is that any method must make use of a considerable amount of memory for the results of intermediate computations. Each interpolation or substitution in an underlying chord sequence obviously requires a memory for the previous state of the chord sequence. Hence a considerable computational power is required to generate tonal chord sequences, and certainly a greater degree than is implicit in the informal writings of music theorists.

Critics are sometimes sceptical about the use of grammars. They argue that people 'often break the rules' in order to produce a more original work of art. They also say that although a grammar may capture a genre, individuals have their own unique styles. Both objections are instructive, but not decisive. If a creative process breaks the rules, then it must either make an arbitrary choice regardless of the consequences or be governed by further criteria. Either way, the end result can be captured in a further grammar. Similarly, if an individual has a unique style, then it must either depend on arbitrary departures from the criteria of the norm or on slightly different criteria. Either way, the end result can be captured in a grammar. Indeed, a grammar can be framed to capture the output of any computational process, whether it is random, probabilistic, or of maximal computational power.

## Conclusions

The case of musical improvisation is an example of creativity within a genre in 'real time', that is, there is no opportunity for revision. The pressure to produce an adequate performance imposes a considerable computational load on the mind. The solution, I have argued, is for the procedure that runs in real time to have a neo-Lamarckian architecture (see Fig. 1). Such a procedure uses all the constraints of the genre in the generative stage of the creative process, and in this way guarantees that the result is always within the genre. The procedure is efficient, and it can also employ an algorithm of weak computational power. The bass and melody programs depend on a long-term memory for chord sequences and for regular grammars of contours and rhythms, but they require only a minimal working memory for intermediate results. They also need a memory that keeps track of the current position in the chord sequence, and a buffer for the most recently produced note. They are psychologically plausible because they would enable a musician to improvise a melody rapidly and without having to carry out complex computations.

Another interesting feature of the programs is that even though they are well understood, it is impossible to predict their output on any particular occasion. They are not deterministic, but they do meet the criteria of the genre, in just the way, I claim, that creative processes characteristically operate.

The creation of chord sequences, and works of art such as novels and paintings, is typically carried out within the conventions of an existing genre. The creative processes of scientists normally occur, as Kuhn (1970) has stressed, within the confines of an existing paradigm. Likewise, the search for counterexamples in everyday reasoning must meet the constraints of the problem. All these types of creativity are likely to depend on a multi-stage architecture—certain constraints are used to generate ideas, and further constraints are used to judge, to monitor, or to revise, the initial products. The activity may be spread over many stages, and it need not occur in real time. Musicians, for example, can write down chord sequences, and work on them in the same way as composers. There is therefore no reason for them to rely on an extensive working memory, unlike my program, because they can consult the written chord sequence. Literacy ensures that computational power exacts no psychological price: notation functions as a memory for the results of intermediate computations. It also enables successive generations of creators to contribute to the development of an idea.

The invention of a new genre or paradigm is the most profound and rarest form of creativity. There can be no neo-Lamarckian algorithm for this type of creativity. In an art such as painting, the revolutions that occur—the invention of perspective, say, or the transition from late Cézanne to Cubism—do not seem to be governed by any common set of constraints. Similarly, in a science such as physics, the shifts in paradigm—the introduction of Newtonian mechanics, say, or the subsequent transition to the theory of relativity—do not seem to be governed by any common underlying principles. The success of new genres and new paradigms may depend on criteria of which their creators are ignorant. These criteria include social and economic factors, and, in the case of science, the results of subsequent empirical studies. If there are no common criteria underlying such revolutions, then it follows that a neo-Lamarckian approach is impossible.

There is a long tradition of proposals that innovation depends on random, or what I have termed 'neo-Darwinian', procedures. Some proposals have been satirical, such as Mozart's scheme for composition by the shake of a dice (see O'Beirne, 1971), or Swift's machine for speculation in the academy of Lagado in *Gulliver's Travels*. Other proposals are serious, and a number of authors have urged that the generation of ideas at random is the only possible creative process (see e.g. Skinner, 1953; Campbell, 1960). The trouble is that the procedure is grossly inefficient. Evolution works because it depends on millions of experiments with billions of organisms over millions of years. As a method for generating ideas in a single head in a single lifetime, it is out of the question—a point that was discovered the hard way, when computer scientists tried to construct intelligent programs by assembling them at random from simple components (Fogel *et al.*, 1966).

The conclusion is inescapable. The search for a profoundly original idea depends on a multi-stage architecture, but it will succeed only if it is guided, at least in part, by constraints of some sort. Knowledge is a potent source of constraints, but knowledge alone is not enough. To return from the profound to the prosaic, everyone recognizes the ingenuity of the post-hypnotic scenario in the case of the murder in the cinema. Nearly everyone has enough knowledge to construct this scenario; yet few people succeed in thinking of it for themselves. Is there perhaps some mental commodity that, if enhanced, would lead to success both here and in other domains—a higher degree of intelligence, a larger working memory, a more rapidly functioning brain, a larger number of associative connections, a higher degree of motivation, or a greater capacity for taking pains? I suspect not. What evidence there is suggests that creativity is not merely a matter of enhancing some such property: there are plenty of highly intelligent and dedicated individuals (by any measure) who lack the spark of originality. My conjecture is that geniuses need to have their knowledge in a form that can directly govern the generative stages of creativity. Conscious critical knowledge, which is relatively easy to acquire, is impotent when it comes to the unconscious generation of ideas. The pedagogical moral is that perhaps the best method to foster creativity is to encourage individuals to attempt to create within a particular domain as soon as they have acquired the rudiments of technique.

## References

Baddeley, A. D. (1966). The capacity for generating information by randomization. *Quarterly Journal of Experimental Psychology,* **18,** 119–129.

Braine, M. D. S. (1978). On the relation between the natural logic of reasoning and standard logic. *Psychological Review,* **85,** 1–21.

Campbell, D. (1960). Blind variation and selective retention in creative thought as in other knowledge processes. *Psychological Review,* **67,** 380–400.

Fogel, L., Owens, A. & Walsh, M. (1966). *Artificial Intelligence through Simulated Evolution.* New York: Wiley.

Forte, A. (1979). *Tonal Harmony in Concept and Practice,* 3rd ed. New York: Holt, Rinehart & Winston.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science,* **7,** 155–170.

Gick, M. L. & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology,* **15,** 1–38.

Henle, M. (1962). On the relation between logic and thinking. *Psychological Review,* **69,** 366–378.

Herink, R. (ed.) (1980). *The Psychotherapy Handbook: The A to Z Guide to More Than 250 Different Therapies in Use Today.* New York: Meridian.

Inhelder, B. & Piaget, J. (1958). *The Growth of Logical Thinking from Childhood to Adolescence.* London: Routledge & Kegan Paul.

Johnson-Laird, P. N. (1975). Models of deduction. In R. J. Falmagne (ed.), *Reasoning: Representation and Process in Children and Adults.* Hillsdale, NJ: Erlbaum.

Johnson-Laird, P. N. (1983). *Mental Models: Towards a Cognitive Science of Language, Inference and Consciousness.* Cambridge: Cambridge University Press. Cambridge, MA: Harvard University Press.

Johnson-Laird, P. N. (in press). Analogy and the exercise of creativity. In A. Ortony (ed.), Proceedings of the Workshop on Analogy at Champaign, Illinois, June 1986.

Kuhn, T. S. (1970). *The Structure of Scientific Revolutions,* 2nd ed. Chicago, IL: University of Chicago Press.

Levitt, D. A. (1981). A melody description system for jazz improvisation. MSc thesis, Department of Electrical Engineering and Computer Science, MIT.

Longuet-Higgins, H. C. (1979). The perception of music. *Proceedings of the Royal Society,* Series B, **205,** 307–322.

O'Beirne, T. H. (1971). From Mozart to the bagpipe, with a small computer. *The Institute of Mathematics and its Applications,* **7,** 11–16.

Osherson, D. N. (1975). Logic and models of logical thinking. In R. J. Falmagne (ed.), *Reasoning: Representation and Process in Children and Adults.* Hillsdale, NJ: Erlbaum.

Perkins, D. N. (1981). *The Mind's Best Work.* Cambridge, MA: Harvard University Press.

Perlman, A. M. & Greenblatt, D. (1981). Miles Davis meets Noam Chomsky: Some observations on jazz improvisation and language structure. In W. Steiner (ed.), *The Sign in Music and Literature.* Austin, TX: University of Texas Press.

Rips, L. J. (1983). Cognitive processes in propositional reasoning. *Psychological Review,* **90,** 38–71.

Skinner, B. F. (1953). *Science and Human Behavior.* New York: Macmillan.

Steedman, M. J. (1982). A generative grammar for jazz chord sequences. Mimeo, Department of Psychology, University of Warwick.

Stravinsky, I. (1947). *Poetics of Music.* London: Faber & Faber.

Sudnow, D. (1978). *Ways of the Hand.* London: Routledge & Kegan Paul.

Sutherland, N. S. (1976). *Breakdown.* London: Weidenfeld & Nicholson.

Ulrich, J. W. (1977). The analysis and synthesis of jazz by computer. *Fifth International Joint Conference on Artificial Intelligence,* 865–872.

---

**P. N. Johnson-Laird** is at the MRC Applied Psychology Unit, 15 Chaucer Road, Cambridge CB2 2EF.

Requests for reprints price £0.50 should be addressed to The British Psychological Society, St Andrews House, 48 Princess Road East, Leicester LE1 7DR.