Agents and **Creativity**

When Alice had finished reciting *You are old, Father William,* at the Caterpillar's request, the following exchange ensued:

"That is not said right," said the Caterpillar. "Not *quite* right, I'm afraid," said Alice, timidly: "Some of the words have got altered." "It is wrong from beginning to end," said the Caterpillar decidedly.

After that, Lewis Carroll tells us, there was silence for some minutes.

The silence was hardly surprising. How can one compare two things—still less judge the closeness of their relationship—if they are different (or "wrong") in *every* respect? Without a series of intermediate structures, one cannot be understood in terms of the other. Even if they do share some similarities, these have to be noticed—and they have to be recognized as significant. Had Alice pointed out that her poem, presumably similar to the one the Caterpillar had in mind, contained several instances of the word "the," the creature would not have been persuaded. Where poems are concerned, metres, rhyming patterns, and many individual words are significant, but "the" counts are not. To argue with the Caterpillar, Alice would have had to identify the important features of her poem, and of his, before being able to compare them.

What has this got to do with creativity, and with agents? Creativity involves coming up with something novel, something different. And this new idea, in order to be interesting, must be intelligible. No matter how different it is, we must be able to understand it in terms of what we knew before. As for agents, their potential uses include helping us by suggesting, identifying, and even evaluating differences between familiar ideas and novel ones. (You'll be relieved, perhaps, if I don't attempt to define just what an agent is—and isn't. Instead, I'll rely on the intuitive notion that an agent is a part of a program which can act, and/or be asked to act, in relative independence of others.)

No one would choose the Caterpillar as an assistant; he

was both grumpy and unhelpful. What Alice wanted to know was just where she had "gone wrong," just where her recital differed from what she had learned before. But the Caterpillar was in no mood to tell her. A computerized agent cannot be grumpy. Whether it can be specifically helpful to someone trying to come up with (or to assess) creative ideas, is what we must consider.

How is creativity possible? How can a person, or a computer for that matter, generate novel ideas? Many scientifically minded individuals would argue that there is nothing especially problematic about creativity. A creative idea, they would say, is merely a novel (and valuable) combination of familiar ideas. Accordingly, creativity could be explained by a scientific theory showing how such novel combinations can come about.

Up to a point, they're right. Samuel Taylor Coleridge's questions about "the hooks and eyes of memory," for instance, could be answered by psychological theories describing the associative processes in the poet's mind (or brain). Indeed, current computer models of neural nets provide some preliminary ideas about just how such associations could happen. Using those ideas, we can lay computational foundations for the *Road to Xanadu* described in a fascinating literary study of the sources of Coleridge's poetic imagery [1, 10].

Moreover, we can see, in outline, how such theories might lead to helpful agents of diverse kinds. For example, the units (on various levels) within a computer model of a rich semantic network could communicate not only with each other but also with a human user. A writer might use the set of intercommunicating agents as an intelligent thesaurus, or even a computerized literary critic. For agents of this type might aid someone stuck for a new image, or someone attempting to assess (or interpret) one suggested by another writer. And they might help to show whether (and how) a series of images fits together, or to diagnose the mixture in a mixed metaphor. An agent within a semantic network could not only effect an association but also trace the associative pathways involved, which in itself might prompt the user to new insights. If such agents were unable always to tell the difference between an interesting image and an inappropriate one, they would be little the worse for that. Human brainstorming sessions, too, produce a lot of rubbish. Nuggets can be found within the rubbish, even though they may require further polishing.

Analogy, too, is the novel combination of familiar ideas. In analogy, the structural similarity between two ideas is especially important. Moreover, the analogy (unlike many associations of ideas) may be systematically developed, for purposes of rhetoric or problem solving.

Several current computer models suggest how two ideas can be seen as analogous, being matched in various ways according to the context of thought [1]. Some are relatively rigid, in the sense that analogies are sought between things having pregiven descriptions, which descriptions are not altered if a new analogy is found [2, 7]. We can think of these programs in agentive terms if we focus separately on the various criteria they use in seeking analogies (semantic, structural, or pragmatic). Interactive versions might be helpful to human writers or problem solvers wanting to find, assess, or compare analogies. (As in the previous discussion of associative agents, this presupposes the availability of a rich database of potentially relevant concepts.)

One analogy model, in particular, is readily seen as a community of agents. The *Copycat* system [6, 11] uses many independent descriptors in trying to interpret a given analogy and to find a new (but similar) one. These descriptors, or "codelets," are applied in parallel, competing with one another to find the strongest analogy. The domain this program works in is very simple: alphabetic letter strings. But there are hidden complexities even in this simple domain. For instance, one and the same structure can be described differently on different occasions, according not only to probabilistic variations but to the specific context surrounding it. Thus, the mini-string *mm* will be described as a letter repetition if it occurs within the larger string *aaffmmppzz*, but as two separate letters (identified respectively as the last and the first member of a successor triplet) if it occurs in the string *abcfghklmmno*. A description can be used for a while and then discarded, as *Copycat* finds a more integrated, high-level, analogy involving other descriptions.

This program has come up with many unexpected alphabetic analogies, some of which are highly persuasive. For example, it was told that *abc* is analogous to (can be changed into) *abd*, and was asked to find a matching analogy for *xyz*. Among the many answers it offered were not only *xyd*, *xyzz*, and *xyy*, but also the surprising and elegant *wyz*. This involves, among other things, mapping *a* onto *z* and *left* onto *right*, and also swapping *successor* and *predecessor*. In embryo, then, we have a set of agents which can say not only "Think of it this way" and "Think of it that way," but also "Better still, think of it like this." If these techniques were made available in an interactive system, they might help human users to see analogies in some unexpected places.

Combinational creativity, then, can be thought of in agentive terms—and might be aided by computer systems made up of many largely independent agents. But is that enough? Can we explain *all* creativity by reference to novel combinations? If not, can we explain other cases in other terms—and would these also be suitable for computer implementation?

Creative ideas include scientific theories; musical compositions; literary genres; instances of choreography, painting, and architecture; theorems of mathematics; and the inventions of engineers. Some of these can be understood as mere novel combinations of familiar ideas. But many cannot, especially those which not only solve the creator's initial problem but also engender a whole new set of problems, to be solved perhaps by the creator's successors over many years. Exploring the implications of a radical new scientific theory, or of a new visual or musical genre, is not a matter of mere combination juggling. On the contrary, it is a structured, disciplined, sometimes even systematic search for the meanings promised.

But how can this be? How can a new idea be pregnant with such promise? Imagine trekking through a desert and up a barren mountainside only to see, from the crest of the hill, a verdant valley stretched out before you. The promise, the possibilities, are enormous. But to find them, you will have to explore the valley — perhaps sketchily at first, but later seeking treasures in many a nook and cranny. Creative thinkers (which means all of us, on a good day) explore the possibilities inherent in their own minds, wherein the spaces are not geographical but conceptual.

A conceptual space is a style of thinking, a mental skill that may be expressed in marble, music, or movement, in poetry, prose, or proof [1]. It is defined by a set of constraints (the dimensions of the space) guiding the generation of ideas in the relevant domain. Some of these constraints are accepted, by the thinker and by the relevant social group (the Caterpillar?), as being more inescapable than others. And some are more fundamental than others. Together, they form a mental landscape with a characteristic structure and potential.

Conceptual spaces are analogous to geographical ones in several ways: they can be mapped, explored, and superficially altered, with many valuable results. In one way, however, they are very different. Unlike physical terrain, a conceptual space can be fundamentally transformed. The result of such a transformation is the appearance of a new space of possibilities, a mental terrain which simply did not exist before. It does not follow that creativity involves only transformations, although many of the most exciting examples do. Many creative achievements result from exploring conceptual spaces in systematic and imaginative ways.

For exploring and transforming our thinking styles—and for understanding and appreciating the results—we need good "maps" of the relevant space. Intuitive maps exist within our heads, mostly inaccessible to consciousness. In more explicit form, they can be found (though usually only in outline) in the humanities: in literary criticism and musicology, in the philosophy of science and aesthetics, and in the history of art, science, and mathematics.

Think of the disciplined beauty of a Palladian villa, for instance, with its symmetrical plan and elegant facade. Or consider the clean lines and interconnecting volumes of a Frank Lloyd Wright open-plan "Prairie House." Think of the conventions of New Orleans jazz, and how they differ from Dizzy Gillespie. And remember how organic chemistry was changed by Kekulé's discovery of the benzene ring, which engendered not just one molecular structure but a vast space of structural potential (aromatic chemistry), whose contents, pathways, and boundaries have now been largely mapped.

Even the Caterpillar might be able to sense the similarity between one Palladian villa and another, or between the various Prairie Houses. And a 20th-century Caterpillar might be able to recognize New Orleans jazz, and distinguish it from later varieties. But Alice's invertebrate friend would not be able to specify the relevant similarities. A well-educated Caterpillar could enumerate the dimensions of the space of benzene derivatives, for these have been made explicit by theoretical chemists. With respect to architecture and jazz, however, things are much less clear.

Generations of architectural historians have disagreed on just what principles of design underlie Palladian design. And an expert on Lloyd Wright's work pronounced the (intuitively evident) architectural balance of his Prairie Houses to be "occult" [8]. However, the crucial stylistic similarities concerned have been explicitly identified within a computer program [3] and a computationally inspired "space-grammar" [8], respectively. Each of these formal systems describes the relevant conceptual space, making it possible to say just why two structures share (or do not share) the same style, and just how (and how fundamentally) they differ.

In addition, each of these systems can generate an indefinite number of structures lying within the relevant conceptual space. Some of these match buildings already designed by Palladio or Lloyd Wright. Others are new, depicting houses of the same general type.

For instance, the plan of a Palladian villa is designed by starting with a rectangle (certain proportions being preferred), and generating internal rectangles (the rooms) by making vertical and horizontal "splits" of various kinds (Palladio himself described this method). Not just any splits will do, if the resulting design is to be one that Palladio would have approved. Splits are unacceptable if they produce internal corridors; long, thin rooms; too many rooms; rooms of greatly disparate size; many internal (windowless) rooms; and the largest rooms lying off the central axis. Early versions of the Palladian program made all of these mistakes, but the relevant design constraints have now been incorporated. Further non-Palladianisms, produced in the past by human imitators (such as Lord Burlington), include rectangular bays jutting out from the rectangular perimeter.

As well as designing plans, this program designs Palladian facades appropriate to a given plan. It knows (among other things) the difference between Doric, Ionic, and Corinthian columns, and the constraints governing the placement of windows and pediment. On several occasions, it has produced a plan-and-facade design virtually identical to one drawn by Palladio himself. In short, the Palladian program, and the Lloyd Wright shape-grammar too, has generated new architectural designs (though not new architectural styles).

The question here, however, is not whether an entire program can do something "creative," or even useful, but whether an agent can help a person to do so. The computational work on architectural styles suggests some ways in which computer agents might help a human architect.

For example, someone designing a Palladian villa, or even combining aspects of Palladianism with some other style, might find it helpful if an informed agent were to suggest—or to forbid—a split at a certain place in the plan. This would be especially useful to architects, or ar-

chitectural students, with little experience of working in this genre. Left to themselves, such a person might design asymmetrical splits, overly narrow rooms, or bays spoiling the clean perimeter line. Similarly, once the house plan had been approved, various agents might offer advice on the facade. Some could suggest the number, and the type, of columns. Others might argue for and against an attic story, or for and against a particular type of pediment or architrave.

The agent initiators and critics could communicate among themselves as necessary (to check for bilateral symmetry, for instance, or for numbers of rooms). But they need not insist on universal agreement. If they are being used as assistants to human beings, then responsibility for the overall integration of the design could be left to the person. However, in some circumstances, the human could not be relied on to achieve a satisfactory design, because of either inexperience or complexity. In producing a set of computerized agents (a program) for practical use, careful judgments would have to be made about just which decisions and evaluations could be left entirely to the user, and which should be monitored, or even made, by the agents. (This is a special case of the familiar problem about the use of "expert systems" in general.)

Agents could be used to help people map, explore, tweak, and (ultimately) transform conceptual spaces of many different kinds. It is not necessary to restrict ourselves to examples such as Palladian architecture, which has long been described in formalist, mathematical, terms. Once a conceptual space has been mapped, agents could help us to move around it in some surprising ways.

Take jazz, for instance. Suppose the Caterpillar in *Alice's Adventures in Wonderland* had not been holding a hookah, but a saxophone. And suppose that Alice, an uncommonly precocious child, had taken some jazz cassettes with her to Wonderland, and played them to the Caterpillar. Could the Caterpillar have been helped to learn to play music in that style by a set of computer agents (also presciently provided by Alice)? Surely, this idea is too nonsensical even for Carroll? The spontaneous creativity of jazz improvisation, you may believe, is simply not the sort of thing computers could help.

Well, that belief might not survive experience of a program that can help people to improvise jazz [4, 5, 14]. This program knows about various dimensions of the musical space of jazz, and various ways of traveling through it. For instance, it can produce fragments of ascending or descending scales, ensuring the scale chosen is the one relevant to the harmony at that particular point. It can provide "call" and "reply" over two or more bars. It can replace the current melody note by another note drawn from the same scale, or provide a chromatic run between this melody note and the next. It can "cut and paste" a library of melodic and rhythmic patterns, or play fractionally ahead of or behind the beat. And it, and the human user, can vary the frequency with which it does any of these things.

If left to wander through the space by itself, this program will improvise—on a given melody, harmony, and rhythm—by making (random) choices on many dimensions simultaneously. Working in this fashion, it often creates novel musical ideas that professional jazz musicians find interesting, and may wish to develop in their own playing. Alternatively, the human user can make the program concentrate on one (or more) dimension at a time, and explore it (or them) in a very simple way. This is why it can help jazz novices, who can focus on the aspect of jazz that is currently causing them difficulty.

The separability of the various musical dimensions suggests the activity of a number of independent musical agents. An improved version of this system might include evaluative modules (agents) which could discriminate between equally legal phrases, or identify weaknesses in an improvisation (its own or the user's) and show how they can be avoided. A really knowledgeable system might even be able to teach the user how to recover from, or even make the best of, a mistake that had not been avoided. (Oliver Sacks has described a patient with Tourette's syndrome who is a jazz drummer, able to turn his unpredictable muscular tics into the seeds of exciting improvisations.) In general, mistakes can be thought of as a sort of serendipity. If knowledgeable agents were developed to help us make the best of our mistakes (not just avoid them), they could lead to some real surprises.

Mention of mistakes raises the question, inevitable in discussions of creativity: "When is a mistake not a mistake?" The answer, sometimes, is "when it is a transformation." Going beyond the familiar conceptual space—generalizing a constraint, specializing it, dropping it, negating it, adding another—could always be described as a mistake, relative to the original style of thinking. Indeed, it often is so described: one of Kekulé's contemporaries dismissed his account of the benzene ring as "a tissue of chemical fancies," and new art forms are commonly rejected when they first appear.

A transformation may be more or less fundamental: changing a string molecule to a ring molecule by closing the formerly open curve, is more fundamental than switching between methyl and ethyl alcohol by making different attachments to the hydroxyl group. And several transformations may be combined: one could include both bays and cylindrical rooms in a basically Palladian design. But not everything can be transformed at once. The new conceptual space is generated from its predecessor, and must be intelligible in certain terms if it is to be accepted. If Alice's recital was literally "wrong from beginning to end," the Caterpillar wouldn't have known what poem she was reciting.

Heuristics for transforming conceptual spaces, including the space of heuristics, have been applied in a number of programs [9]. One of these, whose task is to generate new mathematical concepts from very simple bases, also has criteria for evaluating the mathematical "interest" of the results. Some of the evaluations are mutually exclusive (if the union of two sets either *has* or *does not have* some property possessed by both the original sets, that is counted, rightly, as interesting). Likewise, some

heuristics are opposites: to specialize a concept, and to generalize it, for instance. There is nothing wrong with that, for a concept does not need to satisfy every possible criterion to be interesting, nor does every heuristic need to be applied simultaneously.

These generative heuristics and evaluative rules can all be thought of as agents. Despite being called the "Automatic Mathematician," this program has often been used interactively. For example, a human mathematician would guide a concept into certain pathways by giving it a name, which the program would interpret as a hint to give that concept priority for a while. Versions of this program might be developed for other domains, in which the knowledge and judgment of human users could aid, and be aided by, the application of the transformational and evaluative heuristics. Indeed, its heuristic-altering successor has been applied to several different problem areas, and has generated at least one patentable idea.

The most surprising—though not necessarily the best—transformations would be able to change the conceptual space in unpredictable ways and at unpredictable levels. The clearest example at present is given by genetic algorithms. A crossover operator, for instance, might be thought of as an independent agent that transforms, at random, certain types of constraint represented in the target code.

The targeted constraints may be more or less restrictive. For example, in one graphics program the crossovers and mutations can get right into the heart of the image-generating code [12]. The results are always "viable," in the sense that the newly transformed code will generate some visible image or other. But the process is utterly undisciplined. Although it could be used to help graphic designers come up with images they would never have thought of themselves, it cannot be used to explore or refine an image space in a systematic way. However, that is possible if the mutating agents are allowed to alter only the superficial parameters of the code. Significantly, these less powerful agents are preferred by a professional artist working on "computer sculpture," who uses them to explore specific classes of 3D forms [13].

In both these cases, the evaluation is done by the human user. At each generation, he or she chooses the image or image pair to be used in "breeding" the next generation. In principle, evaluation could be made automatic (in whole or in part) and it might be useful for the artist to be able to avoid having to consider certain sorts of image, or to be presented up front with the most "promising" ones. But if one's interest here is in the development of agent systems for interactive use, evaluation should not be entirely handed over to the computer.

I've concentrated on the practical question of whether agent systems might help to further human creativity. But my discussion can be seen also as an outline of how creativity might be scientifically understood. Many different psychological processes are involved, ranging across combinational and exploratory-transformational thinking. And many questions remain unanswered, or unasked. However, despite all the unclarities, we are beginning to understand the computational resources that underlie creativity in its various forms.

Creativity at Lewis Carroll's level seems magical, but there is no reason to think that it is magic. Wonderland (and the world behind the Looking-Glass, too) owes many of its surprising features to tweakings and transformations of conceptual spaces familiar even to a child. Other Wonderland surprises are grounded in serendipitous associations within the author's mind. We shall never know what all of these were (still less could they have been predicted beforehand). Who can say where the hookah came from? But the Caterpillar and his mushroom may have owed their existence to a real caterpillar and a real mushroom, falling under Carroll's eye on that golden summer afternoon. ◼

References

1. Boden, M. A. *The Creative Mind: Myths and Mechanisms.* (Expanded edition.) Basic Books, New York; Abacus, London, 1991.
2. Falkenhainer, B., Forbus, K.D. and Gentner, D. The structure-mapping engine: Algorithm and examples. *Art. Intell. 41,* 1–63.
3. Hersey, G., and Freedman, R. *Possible Palladian Villas (Plus a Few Instructively Impossible Ones).* MIT Press, Cambridge, Mass, 1992.
4. Hodgson, P. Understanding Computing, Cognition, and Creativity. MSc thesis, University of the West of England, 1990.
5. Hodgson, P. Modelling Cognition in Creative Musical Improvisation (provisional title). DPhil thesis, University of Sussex, in preparation.
6. Hofstadter, D. R., Mitchell, M., French, R., Chalmers, D., and Moser D. *Fluid Concepts and Creative Analogies.* Harvester Wheatsheaf, Hemel Hempstead, in press.
7. Holyoak, K. J. and Thagard, P. Analogical mapping by constraint satisfaction. *Cog. Sci. 13,* 1989, 295–356.
8. Koning, H. and Eizenberg, J. The language of the prairie: Frank Lloyd Wright's prairie houses. *Environ. Plan. B 8,* 1981, 295–323.
9. Lenat, D. B. The role of heuristics in learning by discovery: Three case studies. In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, eds., *Machine Learning: An Artificial Intelligence Approach.* Tioga, Palo Alto, Calif., 1983.
10. Livingston Lowes, J. *The Road to Xanadu: A Study in the Ways of the Imagination.* (2nd edition.) Constable, London, 1951.
11. Mitchell, M. *Analogy-Making as Perception.* MIT Press, Cambridge, Mass., 1993.
12. Sims, K. Artificial evolution for computer graphics, *Comput. Graph. 25,* 4, (Jul. 1991), 319–328.
13. Todd, S. and Latham, W. *Evolutionary Art & Computers.* Academic Press, London, 1992.
14. Waugh, I. Improvisor. *Music Tech.* (Sept. 1992), 70–73.

Margaret A. Boden is a professor of philosophy and psychology in the School of Cognitive and Computing Sciences at the University of Sussex. She is a Fellow of the British Academy, and a Fellow of AAAI. Her research focuses on the philosophical and psychological implications of AI. ***Author's Present address:*** School of Cognitive and Computing Sciences, University of Sussex, Brighton BN1 9QH, England; email: maggie @syma.susx.ac.uk.