EXCERPTED FROM

STEPHEN WOLFRAM
A NEW KIND OF SCIENCE

SECTION 3.11

Some Conclusions

Some Conclusions

In the chapter before this one, we discovered the remarkable fact that even though their underlying rules are extremely simple, certain cellular automata can nevertheless produce behavior of great complexity.

Yet at first, this seems so surprising and so outside our normal experience that we may tend to assume that it must be a consequence of some rare and special feature of cellular automata, and must not occur in other kinds of systems.

For it is certainly true that cellular automata have many special features. All their elements, for example, are always arranged in a rigid array, and are always updated in parallel at each step. And one might think that features like these could be crucial in making it possible to produce complex behavior from simple underlying rules.

But from our study of substitution systems earlier in this chapter we know, for example, that in fact it is not necessary to have elements that are arranged in a rigid array. And from studying mobile automata, we know that updating in parallel is also not critical.

Indeed, I specifically chose the sequence of systems in this chapter to see what would happen when each of the various special features of cellular automata were taken away. And the remarkable conclusion is that in the end none of these features actually matter much at all. For every single type of system in this chapter has ultimately proved capable of producing very much the same kind of complexity that we saw in cellular automata.

So this suggests that in fact the phenomenon of complexity is quite universal—and quite independent of the details of particular systems.

But when in general does complexity occur?

The examples in this chapter suggest that if the rules for a particular system are sufficiently simple, then the system will only ever exhibit purely repetitive behavior. If the rules are slightly more complicated, then nesting will also often appear. But to get complexity in the overall behavior of a system one needs to go beyond some threshold in the complexity of its underlying rules.

The remarkable discovery that we have made, however, is that this threshold is typically extremely low. And indeed in the course of this chapter we have seen that in every single one of the general kinds of systems that we have discussed, it ultimately takes only very simple rules to produce behavior of great complexity.

One might nevertheless have thought that if one were to increase the complexity of the rules, then the behavior one would get would also become correspondingly more complex. But as the pictures on the facing page illustrate, this is not typically what happens.

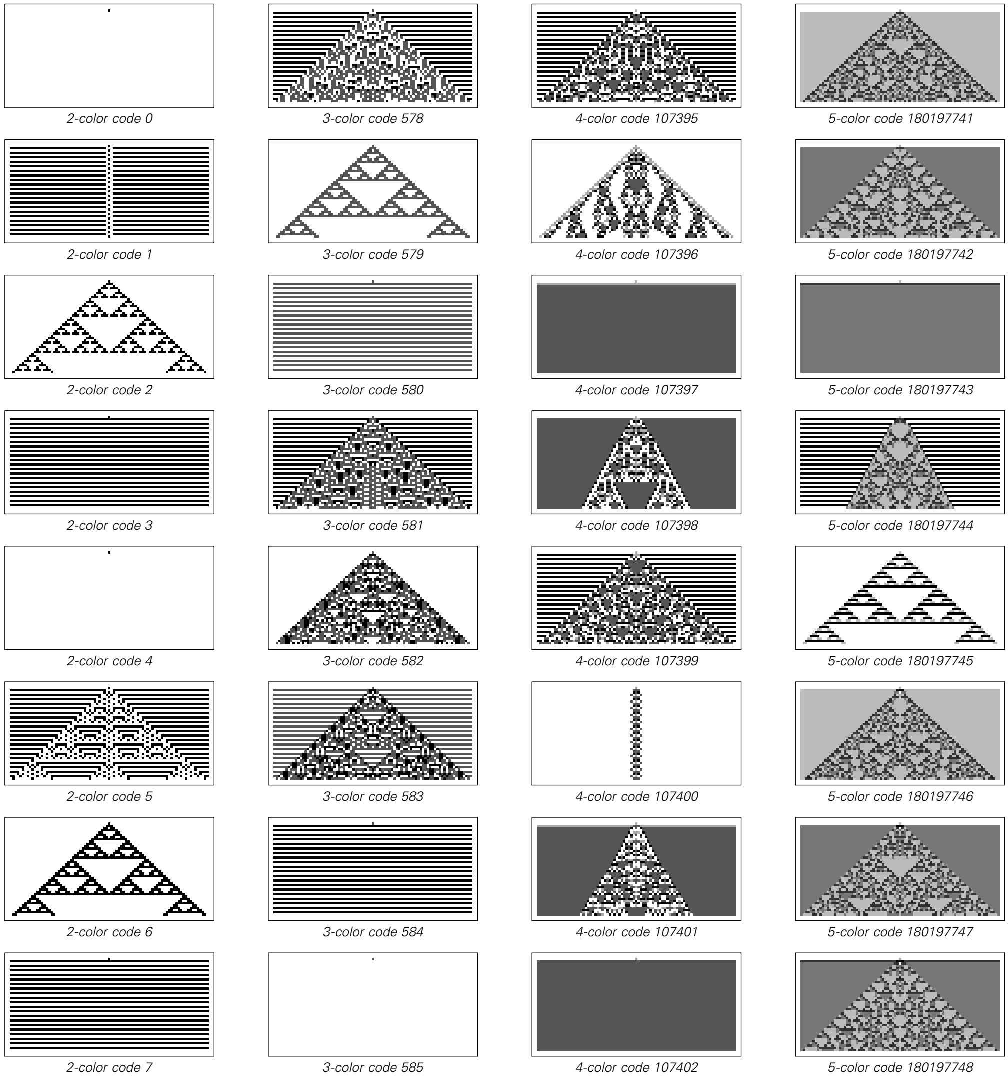
Instead, once the threshold for complex behavior has been reached, what one usually finds is that adding complexity to the underlying rules does not lead to any perceptible increase at all in the overall complexity of the behavior that is produced.

The crucial ingredients that are needed for complex behavior are, it seems, already present in systems with very simple rules, and as a result, nothing fundamentally new typically happens when the rules are made more complex. Indeed, as the picture on the facing page demonstrates, there is often no clear correlation between the complexity of rules and the complexity of behavior they produce. And this means, for example, that even with highly complex rules, very simple behavior still often occurs.

One observation that can be made from the examples in this chapter is that when the behavior of a system does not look complex, it tends to be dominated by either repetition or nesting. And indeed, it seems that the basic themes of repetition, nesting, randomness and localized structures that we already saw in specific cellular automata in the previous chapter are actually very general, and in fact represent the dominant themes in the behavior of a vast range of different systems.

The details of the underlying rules for a specific system can certainly affect the details of the behavior it produces. But what we have seen in this chapter is that at an overall level the typical types of behavior that occur are quite universal, and are almost completely independent of the details of underlying rules.

And this fact has been crucial in my efforts to develop a coherent science of the kind I describe in this book. For it is what implies that



Examples of cellular automata with rules of varying complexity. The rules used are of the so-called totalistic type described on page 60. With two possible colors, just 4 cases need to be specified in such rules, and there are 16 possible rules in all. But as the number of colors increases, the rules rapidly become more complex. With three colors, there are 7 cases to be specified, and 2187 possible rules; with five colors, there are 13 cases to be specified, and 1,220,703,125 possible rules. But even though the underlying rules increase rapidly in complexity, the overall forms of behavior that we see do not change much. With two colors, it turns out that no totalistic rules yield anything other than repetitive or nested behavior. But as soon as three colors are allowed, much more complex behavior is immediately possible. Allowing four or more colors, however, does not further increase the complexity of the behavior, and, as the picture shows, even with five colors, simple repetitive and nested behavior can still occur.

there are general principles that govern the behavior of a wide range of systems, independent of the precise details of each system.

And it is this that means that even if we do not know all the details of what is inside some specific system in nature, we can still potentially make fundamental statements about its overall behavior. Indeed, in most cases, the important features of this behavior will actually turn out to be ones that we have already seen with the various kinds of very simple rules that we have discussed in this chapter.

How the Discoveries in This Chapter Were Made

This chapter—and the last—have described a series of surprising discoveries that I have made about what simple programs typically do. And in making these discoveries I have ended up developing a somewhat new methodology—that I expect will be central to almost any fundamental investigation in the new kind of science that I describe in this book.

Traditional mathematics and the existing theoretical sciences would have suggested using a basic methodology in which one starts from whatever behavior one wants to study, then tries to construct examples that show this behavior. But I am sure that had I used this approach, I would not have got very far. For I would have looked only for types of behavior that I already believed might exist. And in studying cellular automata, this would for example probably have meant that I would only have looked for repetition and nesting.

But what allowed me to discover much more was that I used instead a methodology fundamentally based on doing computer experiments.

In a traditional scientific experiment, one sets up a system in nature and then watches to see how it behaves. And in much the same way, one can set up a program on a computer and then watch how it behaves. And the great advantage of such an experimental approach is that it does not require one to know in advance exactly what kinds of behavior can occur. And this is what makes it possible to discover genuinely new phenomena that one did not expect.

Experience in the traditional experimental sciences might suggest, however, that experiments are somehow always fundamentally imprecise.