EXCERPTED FROM

STEPHEN WOLFRAM

A NEW KIND OF SCIENCE

SECTION 11.7

Implications of Universality

Implications of Universality

When we first discussed cellular automata, Turing machines, substitution systems, register machines and so on in Chapter 3, each of these kinds of systems seemed rather different. But already in Chapter 3 we discovered that at the level of overall behavior, all of them had certain features in common. And now, finally, by thinking in terms of computation, we can begin to see why this might be the case.

The main point, as the previous two sections have demonstrated, is that essentially all of these various kinds of systems—despite their great differences in underlying structure—can ultimately be made to emulate each other.

This is a very remarkable result, and one which will turn out to be crucial to the new kind of science that I develop in this book.

In a sense its most important consequence is that it implies that from a computational point of view a very wide variety of systems, with very different underlying structures, are at some level fundamentally equivalent. For one might have thought that every different kind of system that we discussed for example in Chapter 3 would be able to perform completely different kinds of computations.

But what we have discovered here is that this is not the case. And instead it has turned out that essentially every single one of these systems is ultimately capable of exactly the same kinds of computations.

And among other things, this means that it really does make sense to discuss the notion of computation in purely abstract terms, without referring to any specific type of system. For we now know that it ultimately does not matter what kind of system we use: in the end essentially any kind of system can be programmed to perform the same computations. And so if we study computation at an abstract level, we can expect that the results we get will apply to a very wide range of actual systems.

But it should be emphasized that among systems of any particular type—say cellular automata—not all possible underlying rules are capable of supporting the same kinds of computations.

Indeed, as we saw at the beginning of this chapter, some cellular automata can perform only very simple computations, always yielding for example purely repetitive patterns. But the crucial point is that as one looks at cellular automata with progressively greater computational capabilities, one will eventually pass the threshold of universality. And once past this threshold, the set of computations that can be performed will always be exactly the same.

One might assume that by using more and more sophisticated underlying rules, one would always be able to construct systems with ever greater computational capabilities. But the phenomenon of universality implies that this is not the case, and that as soon as one has passed the threshold of universality, nothing more can in a sense ever be gained.

In fact, once one has a system that is universal, its properties are remarkably independent of the details of its construction. For at least as far as the computations that it can perform are concerned, it does not matter how sophisticated the underlying rules for the system are, or even whether the system is a cellular automaton, a Turing machine, or something else. And as we shall see, this rather remarkable fact forms the basis for explaining many of the observations we made in Chapter 3, and indeed for developing much of the conceptual framework that is needed for the new kind of science in this book.

The Rule 110 Cellular Automaton

In previous sections I have shown that a wide variety of different kinds of systems can in principle be made to exhibit the phenomenon of universality. But how complicated do the underlying rules need to be in a specific case in order actually to achieve universality?

The universal cellular automaton that I described earlier in this chapter had rather complicated underlying rules, involving 19 possible colors for each cell, depending on next-nearest as well as nearest neighbors. But this cellular automaton was specifically constructed so as to make its operation easy to understand. And by not imposing this constraint, one might expect that one would be able to find universal cellular automata that have at least somewhat simpler underlying rules.

Fairly straightforward modifications to the universal cellular automaton shown earlier in this chapter allow one to reduce the number