EXCERPTED FROM

STEPHEN WOLFRAM

A NEW KIND OF SCIENCE

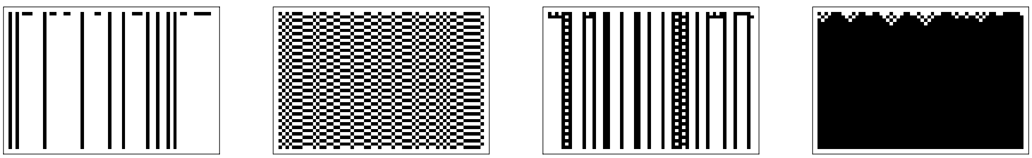
SECTION 11.11

The Threshold of Universality in Cellular Automata

The Threshold of Universality in Cellular Automata

By showing that rule 110 is universal, we have established that universality is possible even among cellular automata with the very simplest kinds of underlying rules. But there remains the question of what is ultimately needed for a cellular automaton—or any other kind of system—to be able to achieve universality.

In general, if a system is to be universal, then this means that by setting up an appropriate choice of initial conditions it is possible to get the system to emulate any type of behavior that can occur in any other system. And as a consequence, cellular automata like the ones in the pictures below are definitely not universal, since they always produce just simple uniform or repetitive patterns of behavior, whatever initial conditions one uses.



rule 4 *rule 51* *rule 108* *rule 250*

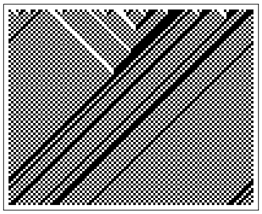
Examples of elementary cellular automata which only ever show purely uniform or purely repetitive behavior, and which therefore definitely cannot be universal. These cellular automata are necessarily all class 1 or class 2 systems.

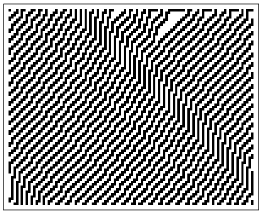
In a sense the fundamental reason for this—as we discussed on page 252—is that such class 1 and class 2 cellular automata never allow any transmission of information except over limited distances. And the result of this is that they can only support processes that involve the correlated action of a limited number of cells.

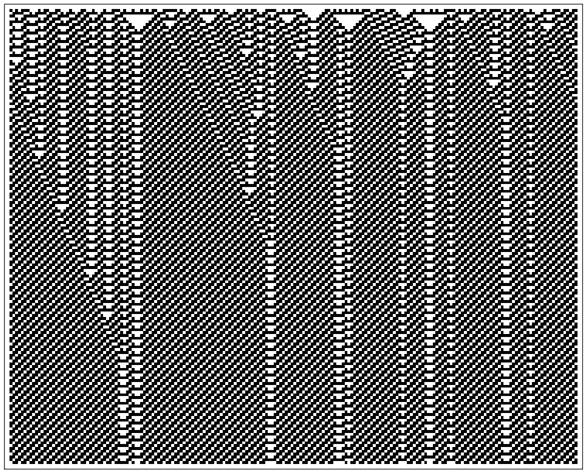
In cellular automata like the ones at the top of the facing page some information can be transmitted over larger distances. But the way this occurs is highly constrained, and in the end these systems can only produce patterns that are in essence purely nested—so that it is again not possible for universality to be achieved.

What about additive rules such as 90 and 150?

With simple initial conditions these rules always yield very regular nested patterns. But with more complicated initial conditions, they produce more complicated patterns of behavior—as the pictures at

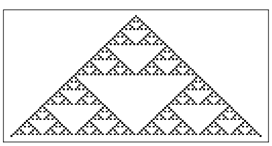

rule 184

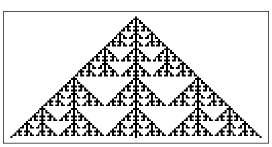

rule 14

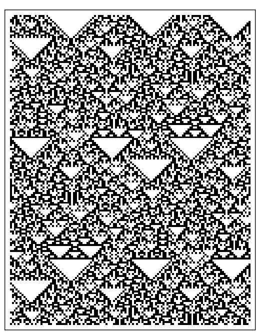

rule 62

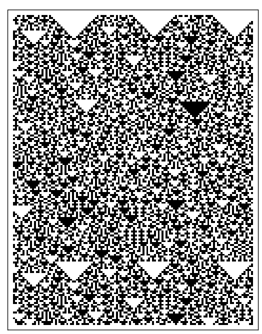
Examples of cellular automata that do allow information to be transmitted over large distances, but only in very restricted ways. The overall patterns produced by such cellular automata are essentially nested. No cellular automata of this kind can ever be universal.

the bottom of this page illustrate. As we saw on page 264, however, these patterns never in fact really correspond to more than rather simple transformations of the initial conditions. Indeed, even after say 1,048,576 steps—or any number of steps that is a power of two—the array of cells produced always turns out to correspond just to a simple superposition of two or three shifted copies of the initial conditions.


rule 90


rule 150

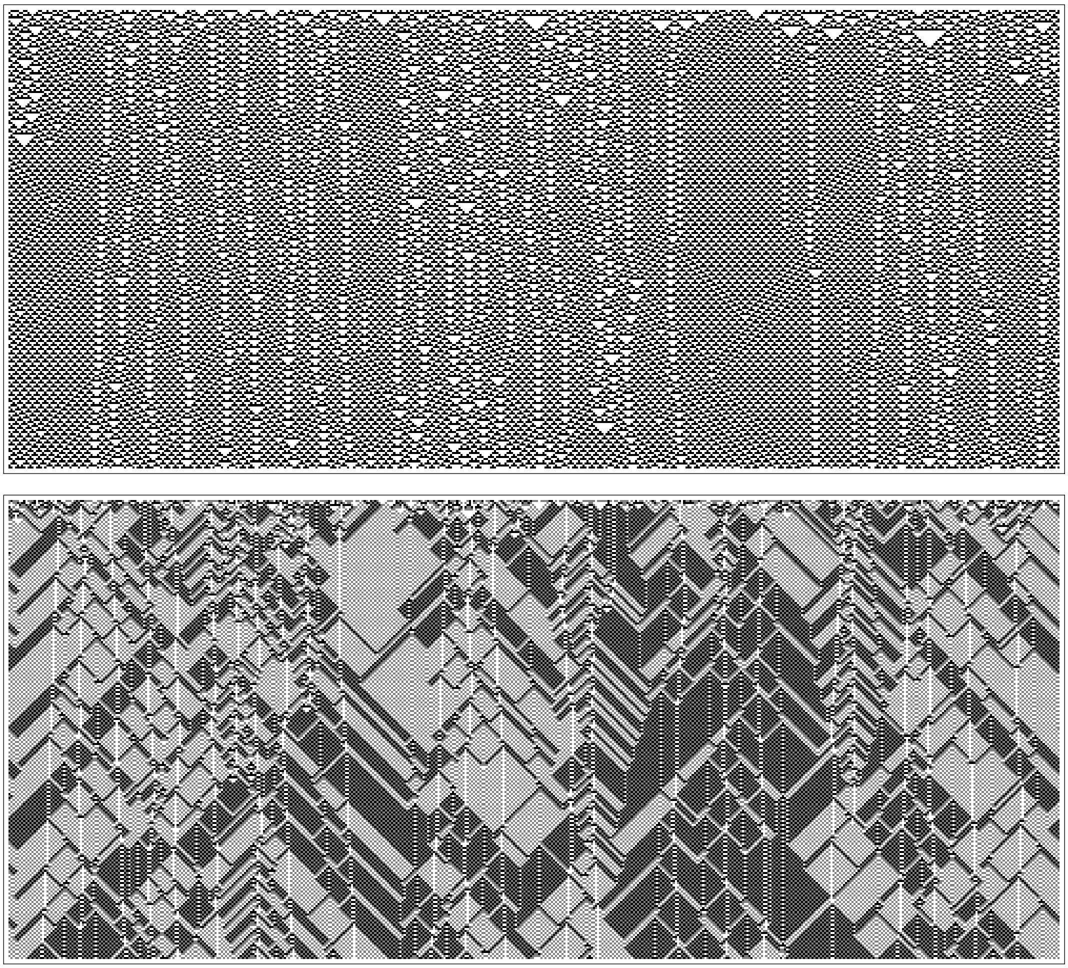

rule 90

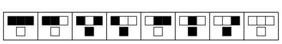

rule 150

Examples of cellular automata with additive rules. The repetitive occurrence of states that correspond to simple transformations of the initial conditions prevent such cellular automata from ever being universal.

And since there are many kinds of behavior that do not return to such predictable forms after any limited number of steps, one must conclude that additive rules cannot be universal.

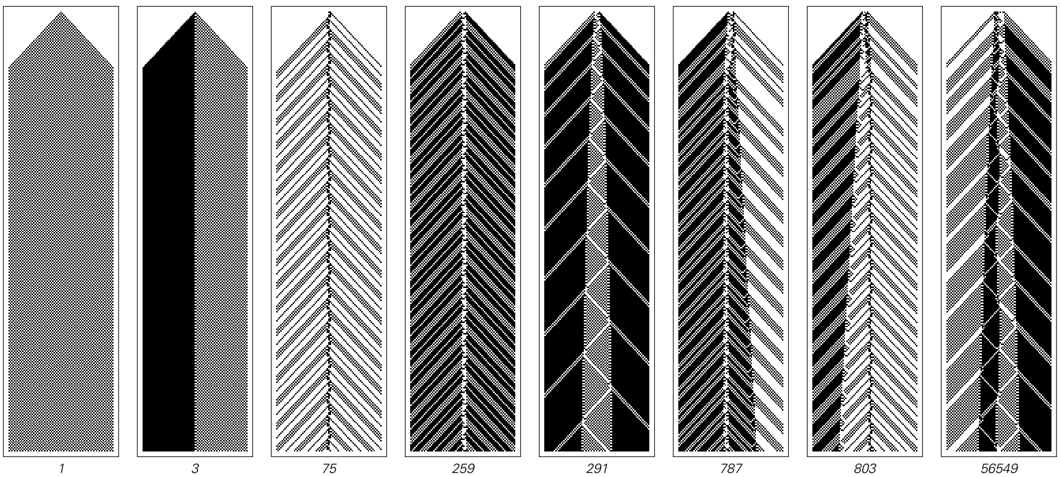
At the end of the last section I mentioned rule 54 as another elementary cellular automaton besides rule 110 that might be class 4. The pictures below show examples of the typical behavior of rule 54.





Two views of the evolution of rule 54 from typical random initial conditions. The top view shows the color of every cell at every step. The bottom groups together pairs of cells, and shows only every other step. There are various localized structures—and hints of class 4 behavior.

Some localized structures are definitely seen. But are they enough to support class 4 behavior and universality? The pictures below show what happens if one starts looking in turn at each of the possible initial conditions for rule 54. At first one sees only simple repetitive behavior. At initial condition 291 one sees a very simple form of nesting. And as one continues one sees various other repetitive and nested forms. But at least up to the hundred millionth initial condition one sees nothing that is fundamentally any more complicated.



Forms of behavior seen in the first 100 million initial conditions for rule 54. With initial condition 291 the n^{th} new stripe on the right is produced at step $2n^2+8n-9$. Even in the last case shown, the arrangement of stripes eventually becomes completely regular, with the n^{th} new stripe being produced at step $n^2+21n/2-\{6, 5, -4, 3\}[[Mod[n, 4]+1]]/2$. Pairs of cells are grouped together in each picture, as at the bottom of the facing page.

So can rule 54 achieve universality? I am not sure. It could be that if one went just a little further in looking at initial conditions one would see more complicated behavior. And it could be that even the structures shown above can be combined to produce all the richness that is needed for universality. But it could also be that whatever one does rule 54 will always in the end just show purely repetitive or nested behavior—which cannot on its own support universality.

What about other elementary cellular automata?

As I will discuss in the next chapter, my general expectation is that more or less any system whose behavior is not somehow fundamentally repetitive or nested will in the end turn out to be universal. But I suspect that this fact will be very much easier to establish for some systems than for others—with rule 110 being one of the easiest cases.

In general what one needs to do in order to prove universality is to find a procedure for setting up initial conditions in one system so as to make it emulate some general class of other systems. And at some level the main challenge is that our experience from programming and engineering tends to provide us with only a limited set of methods for coming up with such a procedure. Typically what we are used to doing is constructing things in stages. Usually we start by building components, and then we progressively assemble these into larger and larger structures. And the point is that at each stage, we need think directly only about the scale of structures that we are currently handling—and not for example about all the pieces that make up these structures.

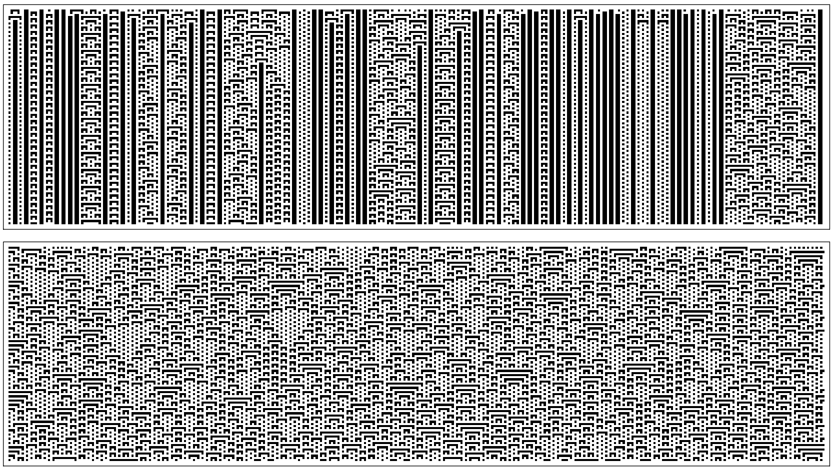
In proving the universality of rule 110, we were able to follow essentially the same basic approach. We started by identifying various localized structures, and then we used these structures as components in building up the progressively larger structures that we needed.

What was in a sense crucial to our approach was therefore that we could readily control the transmission of information in the system. For this is what allowed us to treat different localized structures as being separate and independent objects.

And indeed in any system with class 4 behavior, things will typically always work in more or less the same way. But in class 3 systems they will not. For what usually happens in such systems is that a change made even to a single cell will eventually spread to affect all other cells. And this kind of uncontrolled transmission of information makes it very difficult to identify pieces that could be used as definite components in a construction.

So what can be done in such cases? The most obvious possibility is that one might be able to find special classes of initial conditions in which transmission of information could be controlled. And an example where this can be potentially done is rule 73.

The pictures below show the typical behavior of rule 73—first with completely random initial conditions, and then with initial conditions in which no run of an even number of black squares occurs.

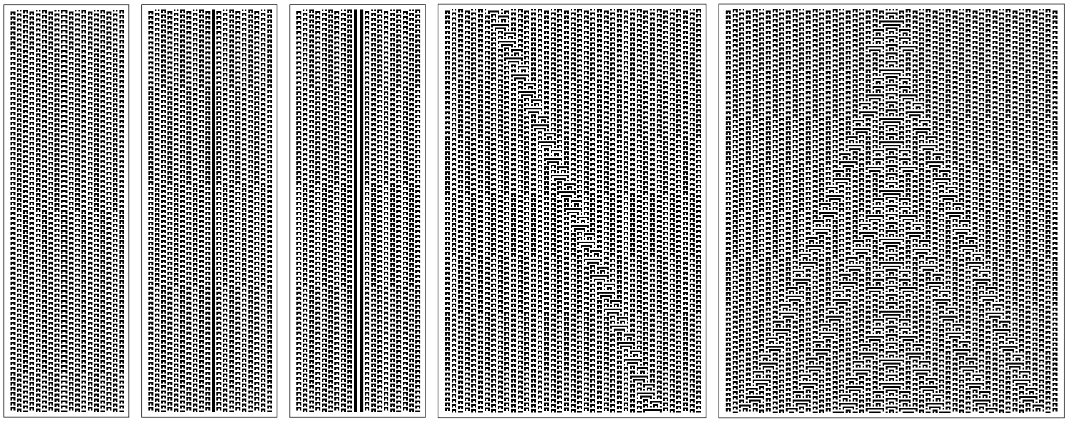


Two examples of rule 73. The top example uses completely random initial conditions; the bottom example uses initial conditions in which no run of an even number of black squares ever occurs. The bottom example is actually part of the pattern obtained from a single black cell—just to the right of the center column, starting with step 1000.

In the second case rule 73 exhibits typical class 3 behavior—with the usual uncontrolled transmission of information. In the first case, however, the black walls that are present seem to prevent any long-range transmission of information at all.

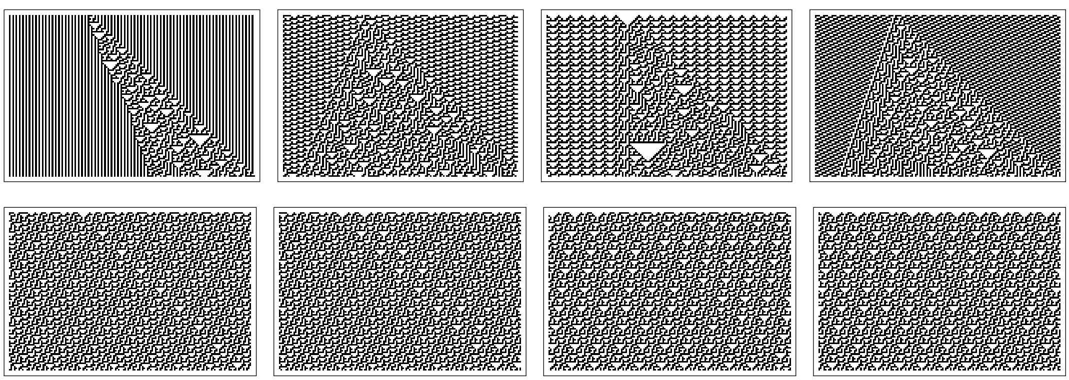
So can one then achieve something intermediate in rule 73—in which information is transmitted, but only in a controlled way?

The pictures at the top of the next page give some indication of how this might be done. For they show that with an appropriate background rule 73 supports various localized structures, some of which move. And while these structures may at first seem more like those in rule 54 than rule 110, I strongly suspect that the complexity of the typical behavior of rule 73 will be reflected in more sophisticated interactions between the structures—and will eventually provide what is needed to allow universality to be demonstrated in much the same way as in rule 110.



Examples of localized structures in rule 73. Note that in the last case shown, the background patterns on either side are mirror images.

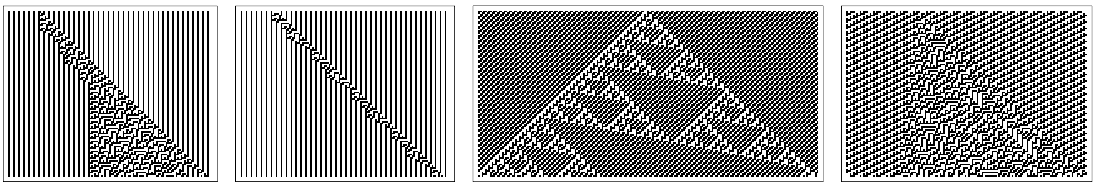
So what about a case like rule 30? With strictly repetitive initial conditions—like any cellular automaton—this must yield purely repetitive behavior. But as soon as one perturbs such initial conditions, one normally seems to get only complicated and seemingly random behavior, as in the top row of pictures below.



Examples of patterns produced by rule 30 with repetitive backgrounds. The top row shows the effect of inserting a single extra black cell into various backgrounds. The bottom row shows all localized structures involving up to 25 cells supported by rule 30 on repetitive backgrounds with blocks of up to 25 cells. Note that these localized structures always move one cell to the right at each step—making it impossible for them to interact in non-trivial ways.

Yet it turns out still to be possible to get localized structures—as the bottom row of pictures above demonstrate. But these structures always seem to move at the same speed, and so can never interact. And even after searching many billions of cases, I have never succeeded in finding any useful set of localized structures in rule 30.

The picture below shows what happens in rule 45. Many possible perturbations to repetitive initial conditions again yield seemingly random behavior. But in one case a nested pattern is produced. And structures that remain localized are now fairly common—but just as in rule 30 always seem to move at the same speed.



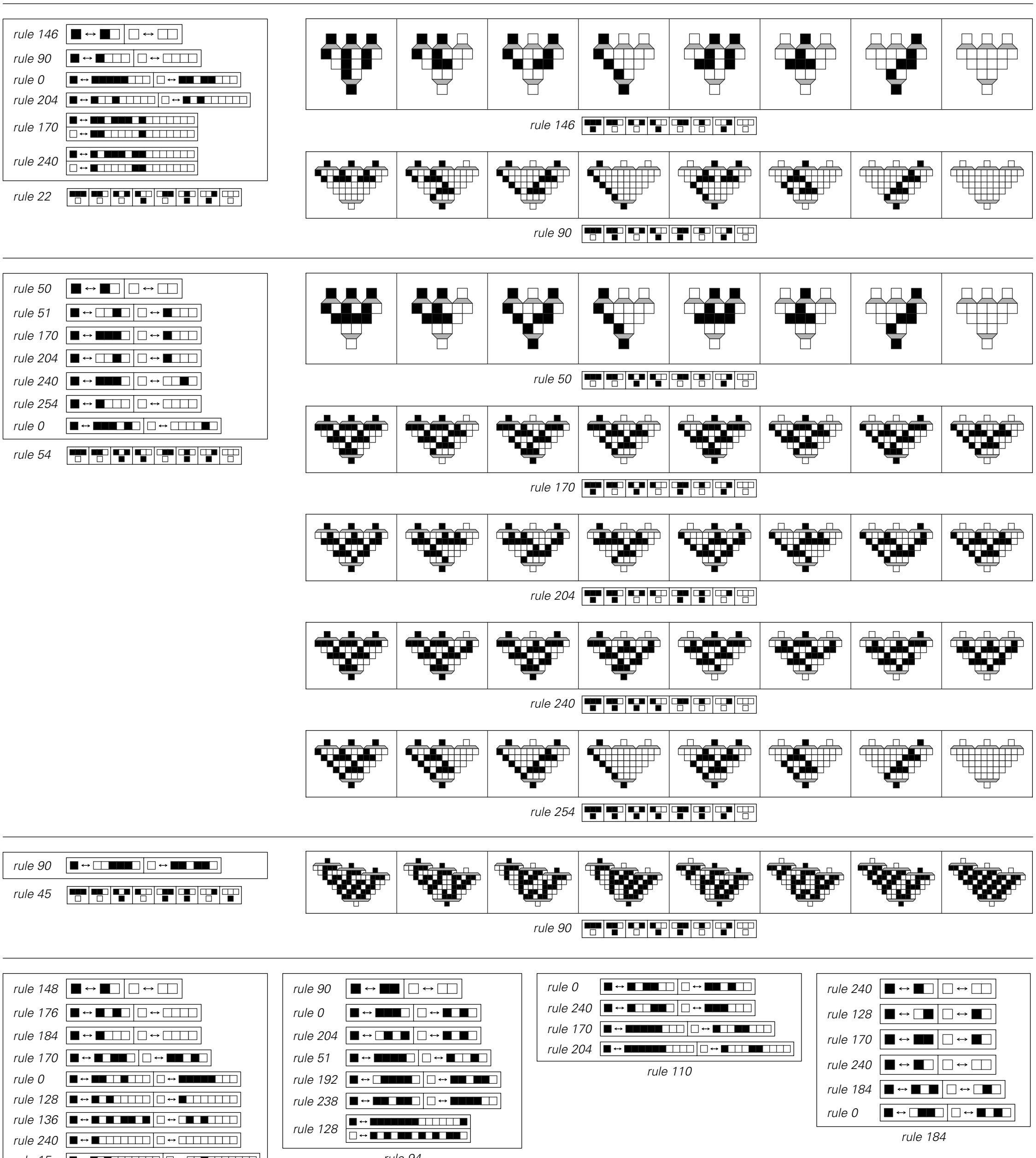
Examples of patterns produced by inserting a single extra black cell into repetitive backgrounds for rule 45. Note the appearance of a slanted version of the nested pattern from rule 90. In rule 45, localized structures turn out to be fairly common—but as in rule 30 they always seem to move at the same speed, and so presumably cannot interact to produce any kind of class 4 behavior.

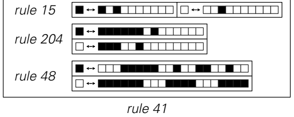
So although this means that the particular type of approach we used to demonstrate the universality of rule 110 cannot immediately be used for rule 30 or rule 45, it certainly does not mean that these rules are not in the end universal. And as I will discuss in the next chapter, it is my very strong belief that in fact they will turn out to be.

So how might we get evidence for this?

If a system is universal, then this means that with a suitable encoding of initial conditions its evolution must emulate the evolution of any other system. So this suggests that one might be able to get evidence about universality just by trying different possible encodings, and then seeing what range of other systems they allow one to emulate.

In the case of the 19-color universal cellular automaton on page 645 it turns out that encodings in which individual black and white cells are represented by particular 20-cell blocks are sufficient to allow the universal cellular automaton to emulate all 256 possible elementary cellular automata—with one step in the evolution of each of these corresponding to 53 steps in the evolution of the original system.





Examples of using various specific elementary cellular automata to emulate other elementary cellular automata. In each case single cells are encoded as blocks of cells, and all distinct such encodings with blocks up to length 20 are shown.

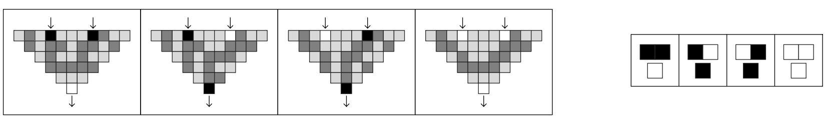
So given a particular elementary cellular automaton one can then ask what other elementary cellular automata it can emulate using blocks up to a certain length.

The pictures on the facing page show a few examples.

The results are not particularly dramatic. No single rule is able to emulate many others—and the rules that are emulated tend to be rather simple. An example of a slight surprise is that rule 45 ends up being able to emulate rule 90. But at least with blocks up to length 25, rule 30 for example is not able to emulate any non-trivial rules at all.

From the proof of universality that we gave it follows that rule 110 must be able to emulate any other elementary cellular automaton with blocks of some size—but with the actual construction we discussed this size will be quite astronomical. And certainly in the picture on the facing page rule 110 does not seem to stand out.

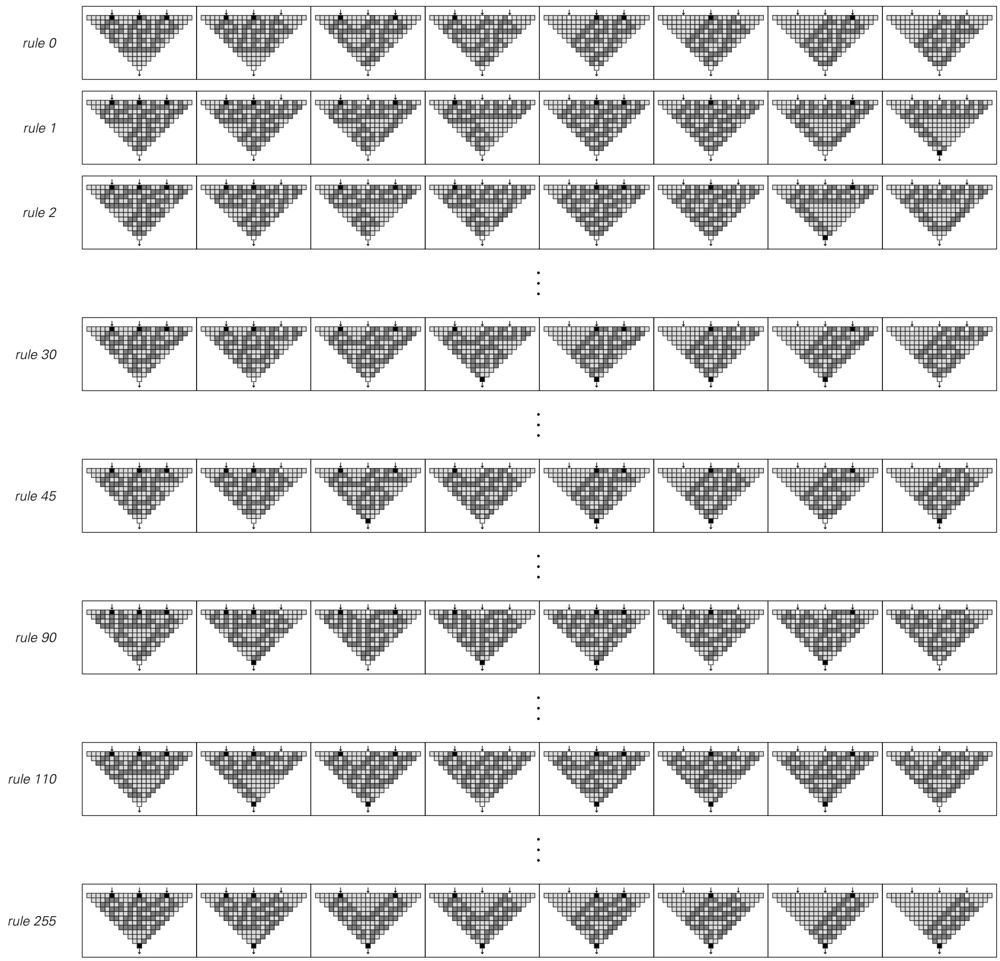
But although it seems somewhat difficult to emulate the complete evolution of one cellular automaton with another, it turns out to be much easier to emulate fragments of evolution for limited numbers of steps. And as an example the picture below shows how rule 30 can be made to emulate the basic action of one step in rule 90.



Rule 30 set up to emulate a single XOR operation—as used in a step of rule 90 evolution. The initial conditions for rule 30 are fixed except at the two positions indicated, where input can effectively be given. The picture shows that for each possible combination of inputs, the result from the rule 30 evolution corresponds exactly to the output from the XOR.

The idea is to set up a configuration in rule 30 so that if one inserts input at particular positions the output from the underlying rule 30 evolution corresponds exactly to what one would get from a single step of rule 90 evolution. And in the particular case shown, this is achieved by having blocks 3 cells wide between each input position.

But as the picture on the next page indicates, by having appropriate blocks 5 cells wide rule 30 can actually be made to emulate

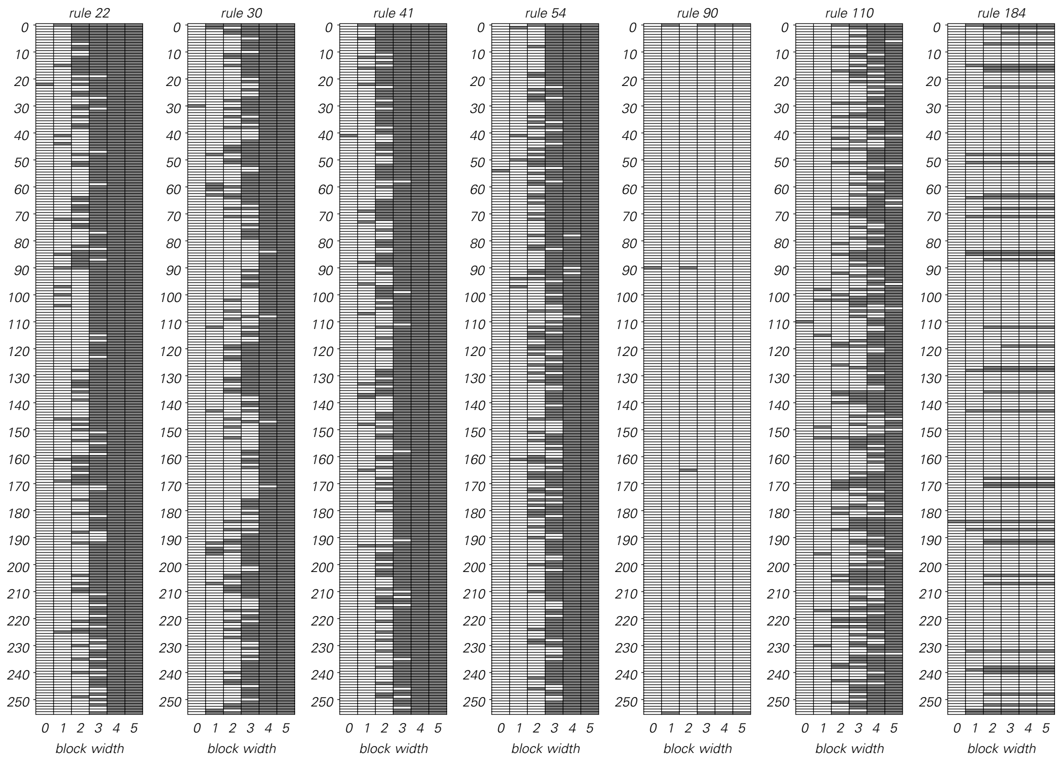


Illustrations of how rule 30 can be set up to emulate a single step in the evolution of all elementary cellular automata.

one step in the evolution of every single one of the 256 possible elementary cellular automata.

So what about other underlying rules?

The picture on the facing page shows for several different underlying rules which of the 256 possible elementary rules can successfully be emulated with successively wider blocks. In cases where the underlying rules have only rather simple behavior—as with rules 90 and 184—it turns out that it is never possible to emulate more than a few of the 256 possible elementary rules. But for underlying rules that have more complex behavior—like rules 22, 30, or 110—it turns out that in the end it is always possible to emulate all 256 elementary rules.



Summaries of how various underlying cellular automata do in emulating a single step in the evolution of each of the 256 possible elementary cellular automata using the scheme from the facing page with blocks of successively greater widths.

The emulation here is, however, only for a single step. So the fact that it is possible does not immediately establish universality in any ordinary sense. But it does once again support the idea that almost any cellular automaton whose behavior seems to us complex can be made to do computations that are in a sense as sophisticated as one wants.

And this suggests that such cellular automata will in the end turn out to be universal—with the result that out of the 256 elementary rules one expects that perhaps as many as 27 will in fact be universal.