EXCERPTED FROM

STEPHEN WOLFRAM

A NEW KIND OF SCIENCE

SECTION 3.3

Mobile Automata

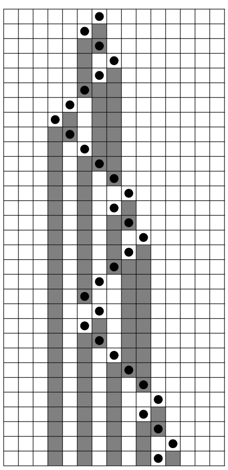
Mobile Automata

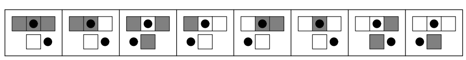
One of the basic features of a cellular automaton is that the colors of all the cells it contains are updated in parallel at every step in its evolution.

But how important is this feature in determining the overall behavior that occurs? To address this question, I consider in this section a class of systems that I call "mobile automata".

Mobile automata are similar to cellular automata except that instead of updating all cells in parallel, they have just a single "active cell" that gets updated at each step—and then they have rules that specify how this active cell should move from one step to the next.

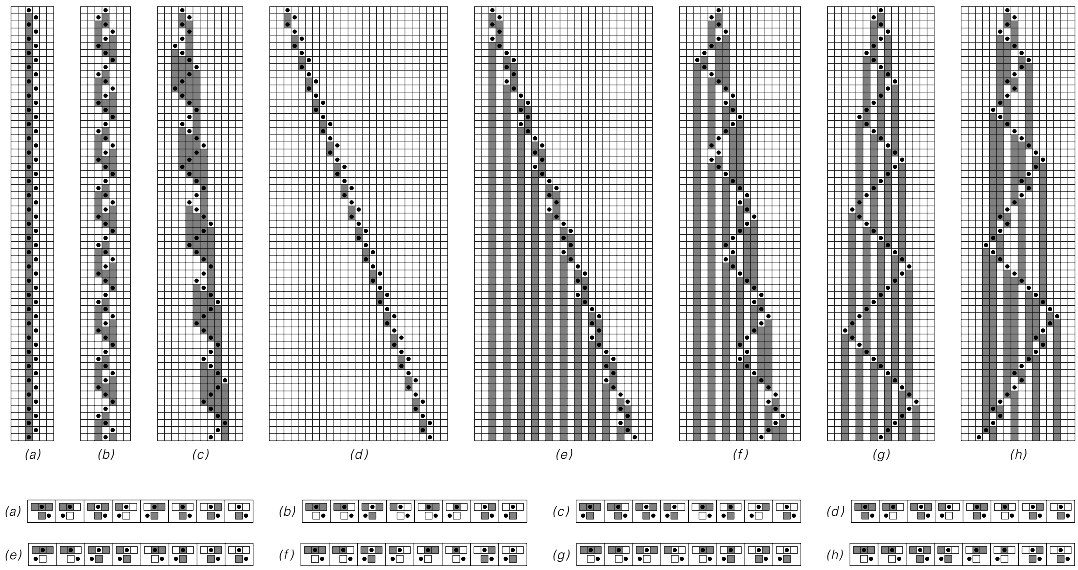
The picture below shows an example of a mobile automaton. The active cell is indicated by a black dot. The rule applies only to this active cell. It looks at the color of the active cell and its immediate neighbors, then specifies what the new color of the active cell should be, and whether the active cell should move left or right.





An example of a mobile automaton. Like a cellular automaton, a mobile automaton consists of a line of cells, with each cell having two possible colors. But unlike a cellular automaton, a mobile automaton has only one "active cell" (indicated here by a black dot) at any particular step. The rule for the mobile automaton specifies both how the color of this active cell should be updated, and whether it should move to the left or right. The result of evolution for a larger number of steps with the particular rule shown here is given as example (f) on the next page.

Much as for cellular automata, one can enumerate all possible rules of this kind; it turns out that there are 65,536 of them. The pictures at the top of the next page show typical behavior obtained with such rules. In cases (a) and (b), the active cell remains localized to a small region, and the behavior is very simple and repetitive. Cases (c) through (f) are similar,

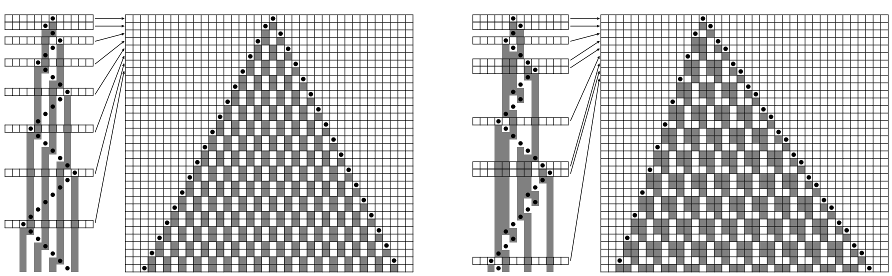


Examples of mobile automata with various rules. In cases (a) through (f) the motion of the active cell is purely repetitive. In cases (g) and (h) it is not. The width of the pattern in these cases after t steps grows roughly like $\sqrt{2t}$.

except that the whole pattern shifts systematically to the right, and in cases (e) and (f) a sequence of stripes is left behind.

But with a total of 218 out of the 65,536 possible rules, one gets somewhat different behavior, as cases (g) and (h) above show. The active cell in these cases does not move in a strictly repetitive way, but instead sweeps backwards and forwards, going progressively further every time.

The overall pattern produced is still quite simple, however. And indeed in the compressed form below, it is purely repetitive.

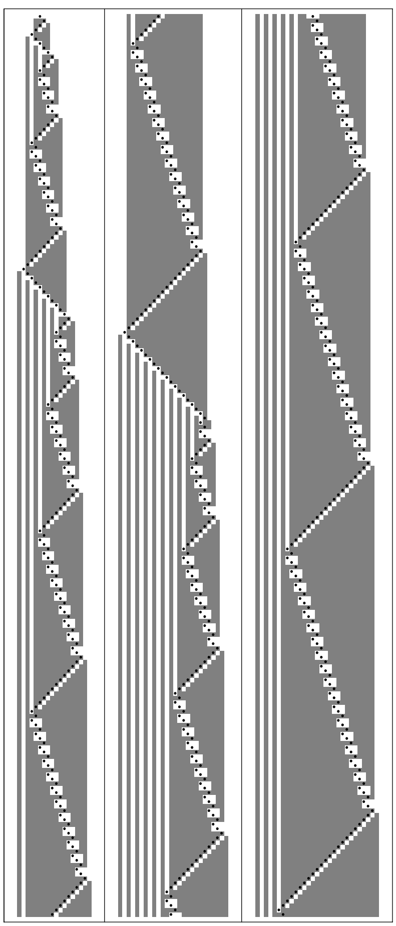


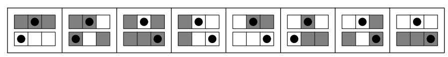
Compressed versions of the evolution of mobile automata (g) and (h) above, obtained by showing only those steps at which the active cell is further to the left or right than it has ever been before.

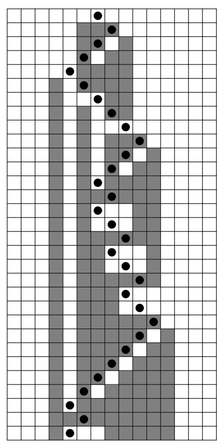
Of the 65,536 possible mobile automata with rules of the kind discussed so far it turns out that not a single one shows more complex behavior. So can such behavior then ever occur in mobile automata?

One can extend the set of rules one considers by allowing not only the color of the active cell itself but also the colors of its immediate neighbors to be updated at each step. And with this extension, there are a total of 4,294,967,296 possible rules.

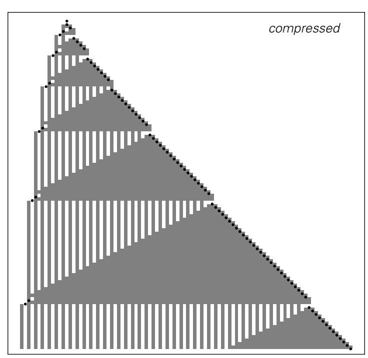
If one samples these rules at random, one finds that more than 99% of them just yield simple repetitive behavior. But once in every few thousand rules, one sees behavior of the kind shown below—that is not purely repetitive, but instead has a kind of nested structure.



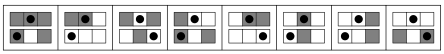




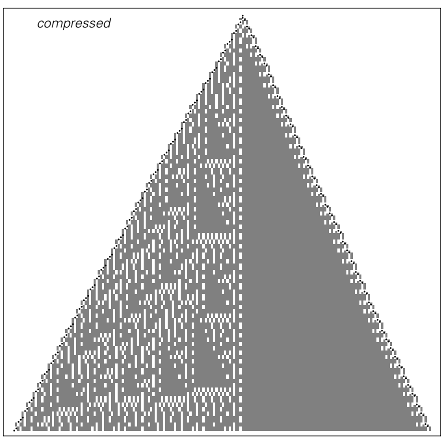
A mobile automaton with slightly more complicated rules that yields a nested pattern. Each column on the left shows 200 steps in the mobile automaton evolution. The compressed form of the pattern is based on a total of 8000 steps.

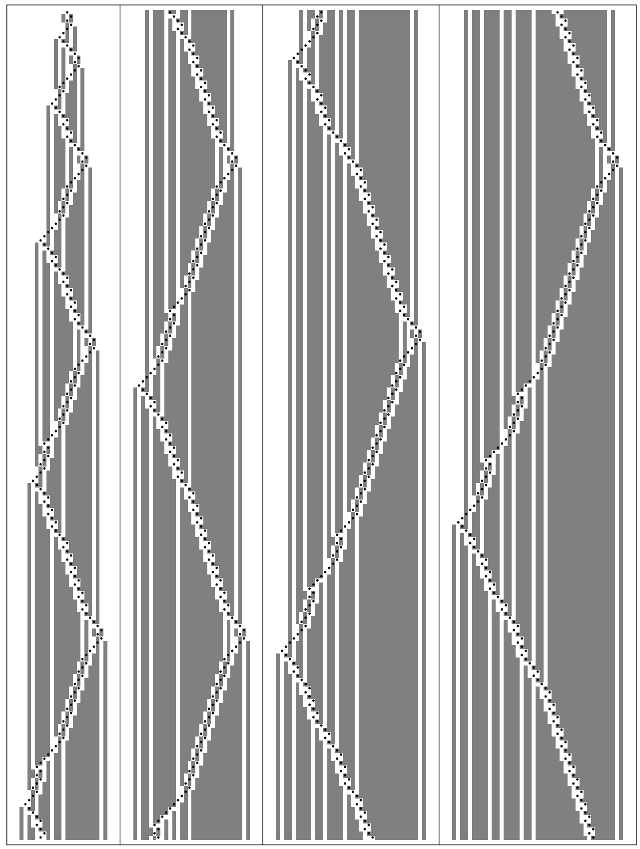


The overall pattern is nevertheless still very regular. But after searching through perhaps 50,000 rules, one finally comes across a rule of the kind shown below—in which the compressed pattern exhibits very much the same kind of apparent randomness that we saw in cellular automata like rule 30.

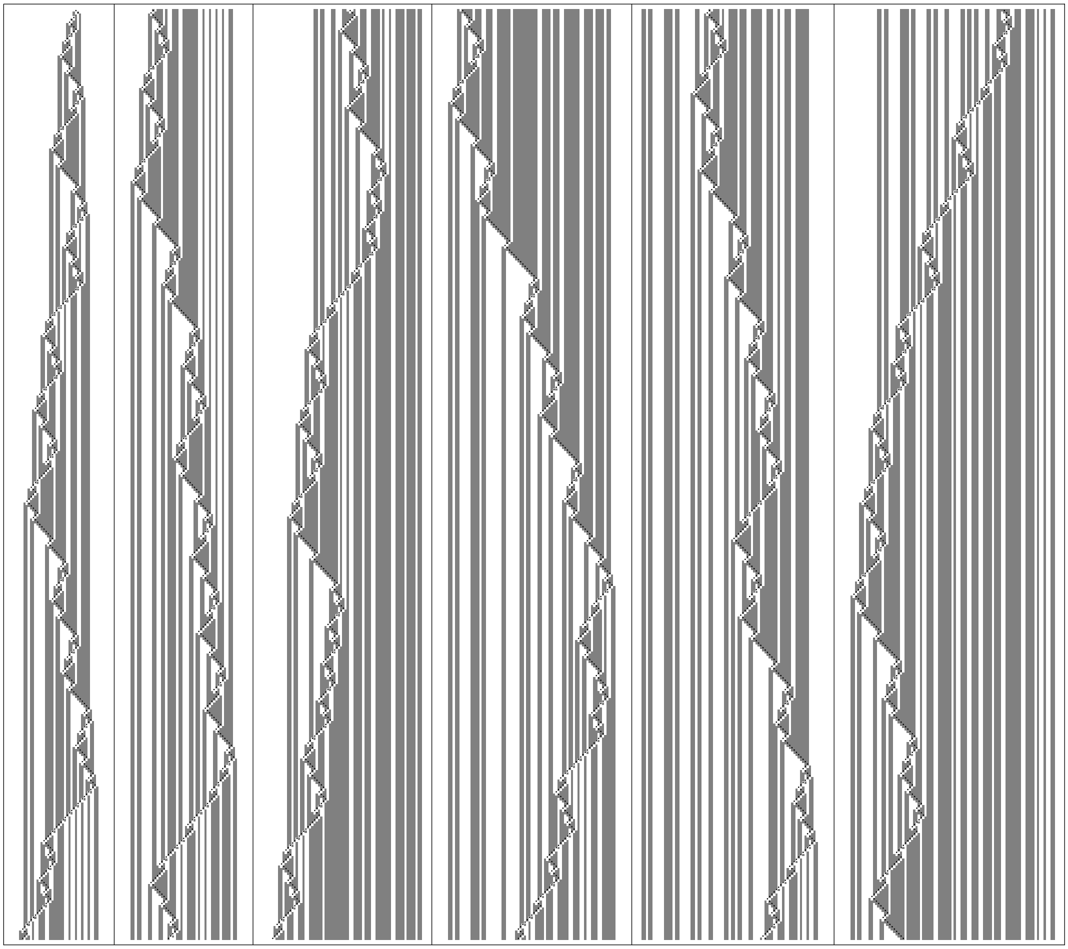


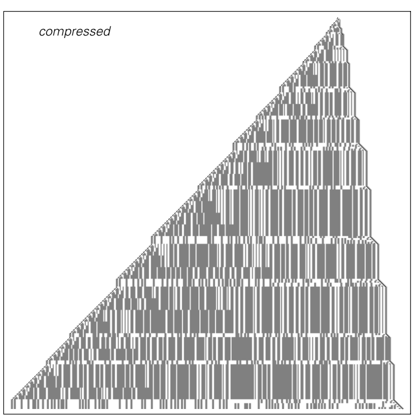
A mobile automaton that yields a pattern with seemingly random features. The motion of the active cell is still quite regular, as the picture on the right shows. But when viewed in compressed form, as below, the overall pattern of colors seems in many respects random. Each column on the right shows 200 steps of evolution; the compressed form below corresponds to 50,000 steps.

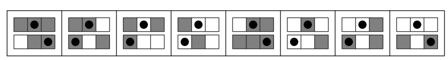




But even though the final pattern left behind by the active cell in the picture above seems in many respects random, the motion of the active cell itself is still quite regular. So are there mobile automata in which the motion of the active cell is also seemingly random? At first, I believed that there might not be. But after searching through a few million rules, I finally found the example shown on the facing page.







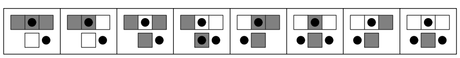
A mobile automaton in which the position of the active cell moves in a seemingly random way. Each column above shows 400 steps; the compressed form corresponds to 50,000 steps. It took searching through a few million mobile automata to find one with behavior as complex as what we see here.

Despite the fact that mobile automata update only one cell at a time, it is thus still possible for them to produce behavior of great complexity. But while we found that such behavior is quite common in cellular automata, what we have seen in this section indicates that it is rather rare in mobile automata.

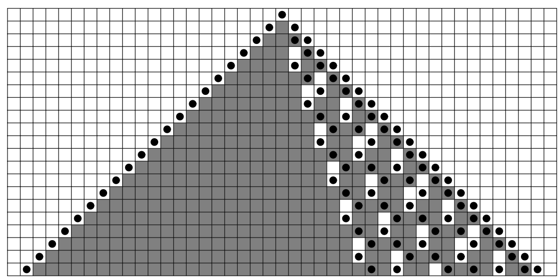
One can get some insight into the origin of this difference by studying a class of generalized mobile automata, that in a sense interpolate between ordinary mobile automata and cellular automata.

The basic idea of such generalized mobile automata is to allow more than one cell to be active at a time. And the underlying rule is then typically set up so that under certain circumstances an active cell can split in two, or can disappear entirely.

Thus in the picture below, for example, new active cells end up being created every few steps.

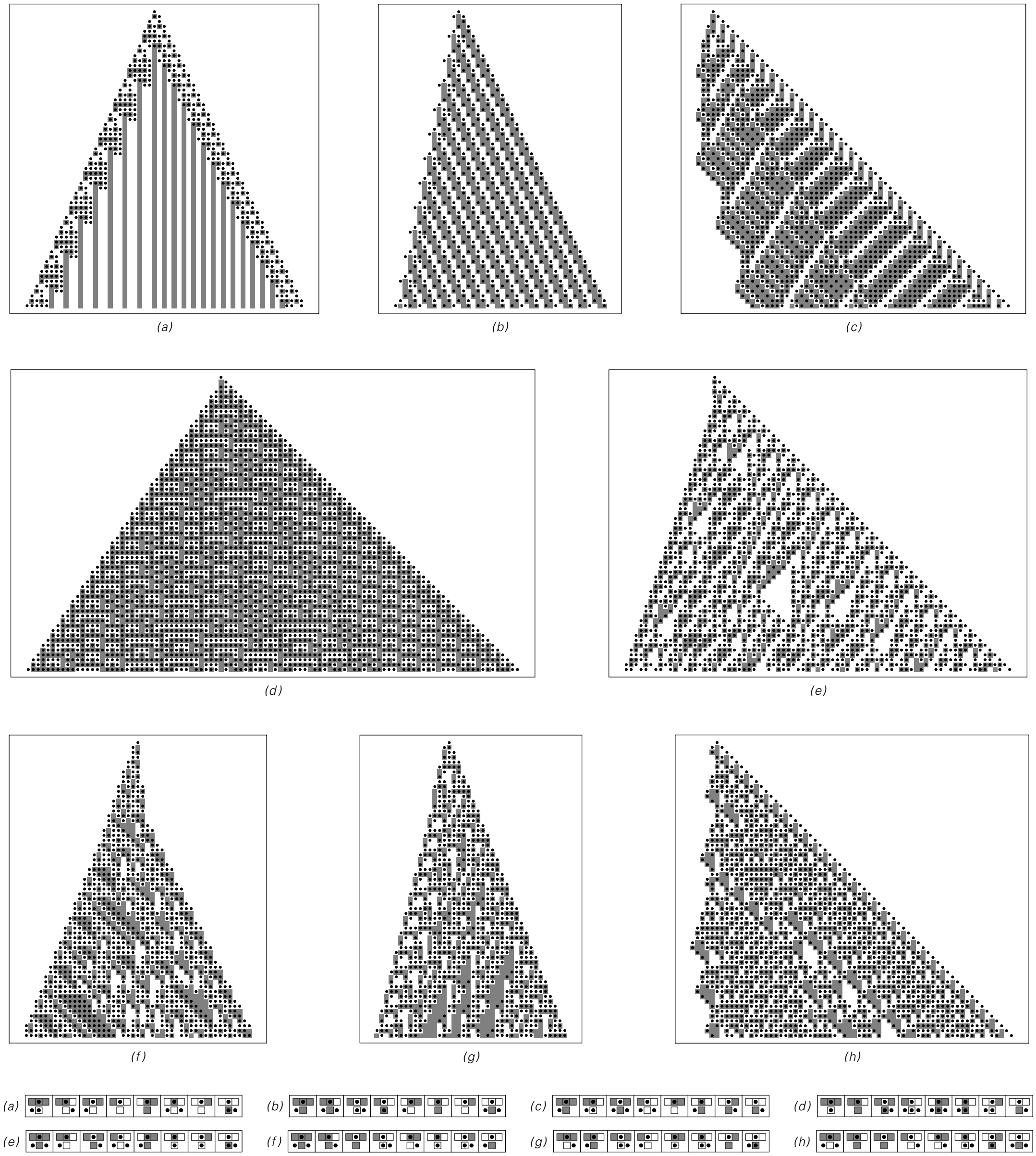


A generalized mobile automaton in which any number of cells can be active at a time. The rule given above is applied to every cell that is active at a particular step. In many cases, the rule specifies just that the active cell should move to the left or right. But in some cases, it specifies that the active cell should split in two, thereby creating an additional active cell.



If one chooses generalized mobile automata at random, most of them will produce simple behavior, as shown in the first few pictures on the facing page. But in a few percent of all cases, the behavior is much more complicated. Often the arrangement of active cells is still quite regular, although sometimes it is not.

But looking at many examples, a certain theme emerges: complex behavior almost never occurs except when large numbers of cells are active at the same time. Indeed there is, it seems, a significant correlation between overall activity and the likelihood of complex behavior. And this is part of why complex behavior is so much more common in cellular automata than in mobile automata.



Examples of generalized mobile automata with various rules. In case (a), only a limited number of cells ever become active. But in all the other cases shown active cells proliferate forever. In case (d), almost all cells are active, and the system operates essentially like a cellular automaton. In the remaining cases somewhat complicated patterns of cells are active. Note that unlike in ordinary mobile automata, examples of complex behavior like those shown here are comparatively easy to find.