Changing Bits to Magnetic Blips

A highly theoretical result from the field of dynamical systems is just what is needed to store data efficiently on computer disks

For nearly 25 years, Roy Alder has done research in theoretical mathematics at IBM's Thomas J. Watson Center in Yorktown Heights, New York. One day, Martin Hassner, an electrical engineer, who is now at IBM in San Jose, came into Adler's office and started talking about the problem of storing data on the magnetic disks of computers. To Adler's amazement, he realized that he himself had the answer to the problem. "I suddenly realized that all these [mathematical] theories that I'd been working on fit [the engineers'] problem like a glove."

Now IBM hopes to start using Adler's theories in its products, although its scientists are guarded in speaking about the company's plans. But they show no hesitation in praising the work. "It is a beautiful example of both serendipity and the applications of mathematics in unexpected ways," says Shmuel Winograd, who is director of mathematical sciences at IBM in Yorktown Heights. "It really exhibits the value of doing mathematics and doing it in quite abstract fields," says Hirsh Cohen, who is vice president for operations of the research division of IBM.

Adler's field is dynamical systems—a mathematical subject that has its origins in concrete physical problems but has been made highly abstract by mathematicians, taking on a life of its own. Dynamical systems, says Adler, began as problems of describing motion, such as the orbits of planets or the paths of falling objects. But mathematicians eventually "abstracted it to the limit" so that it has become a theory of what happens when functions are mapped onto themselves.

A function is a mathematical rule telling what happens to specific points of a set. If you apply a function to a point, you get another point. If you then apply the function to this new point, you get still another point. So, in a dynamical system, when you map a set onto itself, you get a list of points, which the mathematicians call an orbit. These orbits actually are related to physical situations. For example, the path of a particle orbiting in an accelerator can be translated into the language of dynamical systems if you think of the particle as passing through a disk perpendicular to the orbit. Then look at the pattern of intersections of the particle with the disk. The set of intersections is characteristic of the particle's trajectory and if you know which intersection occurred at which time, you can reconstruct the particle's orbit.

Of course, mathematicians who study dynamical systems do not always tie them to such concrete things as orbiting particles. "These things in themselves are objects of study," says Adler. "You can put various mathematical structures on the set. You can put a topology—a notion of distance—on the set and so you can tell when points are close together or far apart. Then you can talk about whether orbits are almost periodic: Do points return to neighborhoods of themselves? On the other hand, you might put a probability measure on the set and ask what's the probability of having an orbit hit a set."

One of the major problems in dynamical systems has been the classification of the systems themselves. Adler explains: "Suppose we have two dynamical systems. Are they really different or are we looking at the same one under different labels? Can you change one system into another by a change of variables?" If so, and if the change of variables also preserves the mathematical structure, then the two systems, which might appear very different, are actually one and the same.

Adler's work was inspired by research on the problem of finding whether probabilistic systems are equivalent. One such system is the mathematical representation of the pattern of heads and tails you get from tossing a coin. Another is the set of faces you get when you roll a die. The question was, Are the two the same? It turns out that they are, as Donald Ornstein of Stanford University discovered, but only if the die is weighted in a certain way.

To get this result, Ornstein used a concept called entropy that is only tenuously related to the well-known entropy of physics. In dynamical systems, entropy measures the complexity of the system. The entropy of the dynamical system that arises from continually tossing a coin and recording whether it comes up heads or tails is log 2. The entropy of continually tossing a die is log 6. To make the dice-tossing system equivalent to the coin-tossing system, you have to load the die so that its entropy is log 2—a task that, although theoretically possible, would take an infinite amount of computer time to actually find the change of variables.

Adler and his associates Benjamin Weiss, who is now at Hebrew University in Jerusalem, and Brian Marcus now at IBM in San Jose were particularly interested in topological dynamical systems. The problem of determining which of these systems are equivalent is similar to the problem of the probabilistic die and coin systems, and they used a concept called topological entropy in their search for a simple way to see if one system can be transformed into another. "We wanted to cut it down to checking a limited number of things," Adler says. "And then we found there's only one thing to check and that's this entropy."

What Adler and his associates found was an "invariant"—a property of the system that would be the same if it were transformed into an equivalent system, so that all they needed to do was check a system's invariant to see if it was the same as another. And, significantly, they found that in topological dynamical systems, their work led directly to an algorithm that transforms equivalent systems into each other. In contrast, when they looked at ergodic, or probabilistic, dynamical systems, they could show only that an appropriate change of variables does or does not exist. They could not find a set of rules for changing equivalent ergodic systems into each other.

At the time, Adler thought that his theories were mathematically interesting but totally without applications. Then Hassner visited him and spoke of the problem of packing data onto magnetic disks. Hassner had read Adler's papers and had noticed the difference between the results for ergodic theory and topological dynamics. He thought the work on topological systems might have applications because it involved a specific algorithm. At that time, says Adler, "For the most part, and especially in connection with this entropy, I'd never paid too much attention to what was happening around me at IBM because I was just working on this abstract mathematics." Moreover, he remarks, the problem Hassner described really had to be pointed out to him as a possible application of his work. "It I'd been put on [the engineers'] problem to solve, I'd never have been able to do it."

Engineers are faced with an enormous amount of data going into and coming out of computers. These computer data are represented by 0's and 1's. The engineers must transform these bits of data into magnetic signals so that they can be stored on disks.

The problem is that there are restrictions on how the computer data can be encoded on magnetic disks and still be read accurately. Only certain numbers of 0's or 1's can follow each other consecutively. For example, says Winograd, in some situations there must be at least three but no more than ten 0's between 1's. The reason is that if 0's and 1's follow each other too closely. the magnetic flux changes too fast and if there is too long a segment of 0's, the flux does not change quickly enough for the reading head to maintain its accuracy. "Engineers are afraid both of losing synchrony and of magnetic drift," Winograd remarks.

But the computer data are essentially random streams of 0's and 1's. How, then, can you transform them into sequences that obey the restrictions of the magnetic disks in the most efficient way and also have an easy rule for transforming them back into computer bits again?

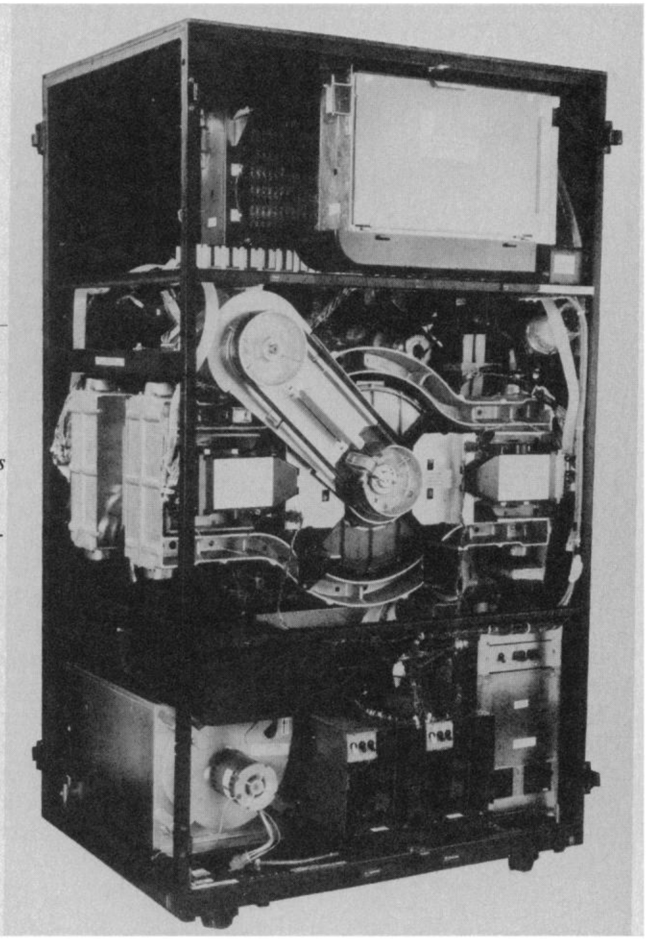
Any method will necessarily have to make the string of computer bits longer when it is encoded onto the magnetic disk. In order to obey the constraints, extra "nonsense" bits will have to be added and rules for adding these extra bits will have to be so precise that the computer can easily remove them again when it reads the disk and converts the magnetic signals into bits again. One rule, for example, might be to add three 0's and a 1 every time you see a 1 and to add a 1 after every string of ten 0's. But that sort of rule, says Adler, is just so incredibly inefficient that it would not even be considered.

And the longer the computer message, the worse the situation because there are so many more combinations of 0's and 1's that are possible that it becomes extremely difficult to make rules to take all these possible sequences into account. A sequence of only ten 1's and 0's already has 2^{10} possible arrangements. The actual sequences are so incredibly long that they can't be collected whole into the memory of the encoding chip but must instead be read piece by piece and continually transformed according to some well worked out rules.

Now, says Winograd, "comes the beautiful observation. Think of the computer data as an infinite stream of 0's and 1's. Then you have a dynamical system." So the problem becomes one of changing the dynamical system you get from the computer into an equivalent dynamical system that obeys the constraints of the magnetic disk. This is exactly the problem that Adler and his colleagues had worked on. It was a topological dynamical system and so was a system for which they had an algorithm. Their theories told not only what was the best possible encoding on magnetic disk but also exactly how to do it. "We have a constructive method to solve the problem. All you need to know are the constraints," says Adler.

A modern high density disk file

In the IBM 3380 computer, the disk file is in the center, looking somewhat like a phonograph. The disks are stacked and are turned by a spindle. A reading arm moves in and out, converting the magnetic signals on the disk into computer bits, represented by 0's and 1's. This disk was not encoded by the new system and IBM is not yet ready to say which disks will be.



The theories of Adler and his colleagues were not immediately ready for this engineering application, but essentially the results were there. "It took a little adjustment," says Adler. "We had to do some revamping of our previous work." But now he has the mathematics at a stage in which "you can put it on a computer, push a button, and in less than a minute you will get the result." In fact, he remarks, the results are so good that "we almost get a circuit diagram out of it."

It is not as though engineers had been unable to encode data onto magnetic disks because of the complex mathematical problem involved. In fact, some of the kinds of codes that Adler produced had been discovered independently by engineers using these empirical methods. But they had had to work long and hard each time to find good encoding methods. "They were extremely clever," says Adler. "But up until now, all they had were ad hoc methods and a bit of luck."

Adler has gone over all the encoding methods that engineers have used and, in each case, he sees how the same methods would arise—and much more easily—with his algorithm. "Armed with the mathematics, it's really pretty easy," he remarks. "But I don't see how those guys did it before."

Cohen says that what particularly pleases him about the ties between Adler's theories and the encoding problem is the history of the work. "Dynamical systems came out of physics, but as mathematicians studied them they became more and more abstract. Now they are coming back into use again and we are benefiting from all the mathematical depth. We have made a connection with a deep mathematical theory and a whole new realm opens up." And the practical benefits for engineers, Cohen adds, are profound. "It is like in medicine when you suddenly understand the mechanism of a disease. Then you no longer have to do palliative things."—GINA KOLATA