Separating the Unsolvable and the Merely Difficult

By GEORGE JOHNSON

Anyone trying to cast a play or plan a social event has come face-to-face with what scientists call a satisfiability problem. Suppose that a theatrical director feels obligated to cast either his ingénue, Actress Alvarez, or his nephew, Actor Cohen, in a production. But Miss Alvarez won't be in a play with Mr. Cohen (her former lover), and she demands that the cast include her new flame, Actor Davenport. The producer, with her own favors to repay, insists that Actor Branislavsky have a part. But Mr. Branislavsky won't be in any play with Miss Alvarez or Mr. Davenport.

As more roles need to be filled, the problem becomes harder and harder to solve. Is it possible to satisfy the tangled web of conflicting demands? These satisfiability problems, called SAT problems for short, arise in thousands of situations, from staffing companies and scheduling airline flights to planning a wedding dinner that won't devolve into a food fight.

And, reaching beyond such practical considerations, researchers embrace satisfiability problems as a tool for studying a phenomenon called computational complexity: some problems are inherently easy to solve, some difficult and some impossible, but scientists are only beginning to understand why.

A paper in the current issue of Nature adds to a string of developments on the topic that have emerged over the last few years. In it an international team of physicists and computer scientists describes work that could help weed out problems that are hopeless and suggest more efficient methods, or algorithms, for solving the rest.

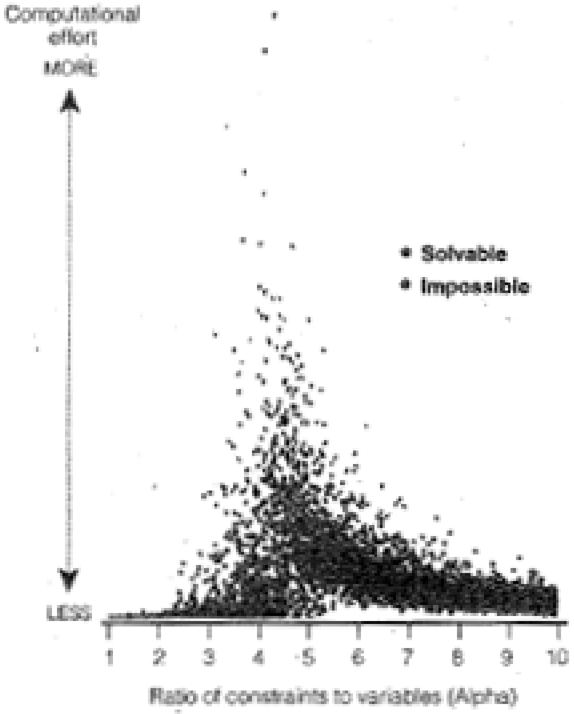
Most intriguingly, this and recent research from other laboratories may help clarify a curious phenomenon that has puzzled scientists for years: the existence of a kind of "phase transition" in which satisfiability problems suddenly change from easy to hard as abruptly as water freezing into ice. A deeper understanding of this parallel might allow physicists to apply ideas about phase transitions occurring in nature to understanding complex mathematical problems, and vice versa: the computer scientists might help shed light on phenomena in the physical world.

"We hope we have finally found the right point of contact between physics insights and computational complexity," said Scott Kirkpatrick of the I.B.M. Thomas J. Watson Research Center in Yorktown Heights, N.Y. He is a co-author of the most recent study, which also included Rémi Monasson of the

Continued on Page 4

Solving Problems

Imagine trying to cast 35 temperamental actors in a play. Some of them hate one another and won't be in the same production. Others will participate only if their friends are also hired. The result is called a satisfiability problem. Depending on the set of constraints that must be met, such problems can be easy, difficult or impossible. To get a grip on the situation, scientists plot various instances (think of each as a different play to be cast) on a graph. As a parameter called alpha is increased, the problems suddenly change from easy to impossible — like water abruptly freezing into ice. In the middle are the most tantalizing problems: those that are difficult but not hopeless.



Source: American Scientist

The New York Times

Separating Insolvable and Difficult

Continued From First Science Page

CNRS-Laboratoire de Physique Théorique in Paris; Riccardo Zecchina of the Abdus Salam International Center for Theoretical Physics in Trieste, Italy; Bart Selman of Cornell University, and Lidror Troyansky of Hebrew University in Jerusalem.

"This connection between such seemingly different fields is likely to shed light on both," said Tad Hogg of the Xerox Palo Alto Research Center in California, who has done pioneering work in this area with his colleague Bernardo Huberman. "Practically, it may lead to better search algorithms for typical problems, though that remains to be seen."

In computer science, problems can be rated in difficulty by how much time it takes to search for solutions. If there are just a few actors to cast in a play, the answer can be found simply by trial and error. For five actors, there are only 2 to the 5th possibilities, a mere 32 combinations: A is in, B is out, C is out, D is in, E is out. The director can try every combination and see if any are consistent with everyone's demands.

In practice, it is usually not necessary to "exhaustively search the problem space," as mathematicians say. It is quickly evident that there is no way to cast Miss Alvarez, but that the play can go forward with Mr. Cohen. For enormous problems involving many variables, pursuing such shortcuts can often save valuable computer time.

Viewed more abstractly, the casting example belongs to a class called 2-SAT problems. If parsed into a logical formula (a kind of mathematical syntax called "conjunctive normal form"), they contain so-called clauses like "Alvarez or Cohen," which each involve no more than two of the potential actors. If one tries to cast a play with 10 or 100 actors, the problem becomes harder. But as long as there are no more than two variables, or actors, in each of the many clauses, the solution time increases relatively slowly, in what is called polynomial time. Thus 2-SAT problems are said belong to a class of solvable problems called P.

But in more complex situations, with clauses that each have at least three variables ("Alvarez or Branislavsky or Cohen"), the space of possible solutions to explore can explode exponentially as the problem grows larger, with more actors added to the play. In the worst cases, these 3-SAT problems — even ones with only 50 actors — rapidly become unsolvable, even given eons of time. This places them in the domain of difficult problems called NP-complete. You can guess at the answer and quickly determine if it's right or wrong (by coming up with a combination of actors and seeing if it works.) But systematically solving the problem can essentially take forever.

The initials NP stand for the rather opaque term "nondeterministic polynomial." The important point is that all NP-complete problems are intimately related. Other examples include the famous traveling salesman problem, in which you try to find the shortest route connecting many different cities. As the number of destinations increases, the difficulty can rise exponentially, and even good approximations are a challenge. If a mathematician could find a general means of solving satisfiability problems this would also dispose of the traveling salesman problem and thousands of others.

But mathematicians strongly suspect that unless they have been missing something, there is no general way to solve large NP-complete problems precisely during the lifetime of the human race or even the universe. Instead, researchers look for ways to understand which problems are merely difficult and which are impossibly complex.

If all 2-SAT problems were solvable and all 3-SAT problems were not, life would be easy — just don't bother with 3-SAT's. But the situation is not so clear-cut. A class of problems is relegated to NP-complete if even a single case is intractable. Though the constraints involved in casting one play might cause the problem to grow so rapidly in complexity that no computer could solve it, a similar play might be cast with relative ease. The challenge is to find a way to separate the two.

Examining the secrets of computational complexity.

Computer scientists start by randomly generating a pool of SAT problems each involving different actors and different casting constraints. (Or the variables might be thought of as airplanes, and the constraints as the routes they must fly.) Then each problem is located as a dot on a graph. A variable called alpha (the ratio of the number of clauses, or demands, to the number of actors) runs along the horizontal axis and a measure of the difficulty of solving the problem on the vertical axis. Then scientists study the way the problems cluster into patterns.

On the surface, the outcome is easily predictable: If the alpha ratio is low, there are only a few demands to satisfy and a large supply of actors to draw on. Solutions tend to be plentiful. If, on the other hand, the ratio of demands to available actors is high enough, mutually agreeable solutions are probably impossible.

The surprise comes from looking deeper: One might expect that the transition from easy to unsolvable would be gradual. But there is an abrupt transition analogous to water freezing into ice as the thermometer drops from 33 to 32 degrees Fahrenheit. A tiny increase in the parameter called alpha causes the problems to suddenly become very hard.

And studies have shown an even more intriguing pattern. The easy problems to the left of the divide can obviously be solved quickly. This is also true for the problems on the other side of the divide: It usually becomes quickly evident that they are impossible and can be abandoned. It is the barely solvable problems lying in the region in between — in the zone of the so-called phase transition — that are the most time-consuming. They are hard but not obviously futile.

Whether one is seeking the most efficient paths for the circuitry on a computer chip or scheduling airplanes, it is in this nether region where the most tantalizing problems lie. "That's why there is such strong interest in what happens at this phase transition," said Dr. Selman, one of the authors of the Nature report. "It's where a lot of real problems are going to be."

For a while, some computer scientists held out hope that all problems that showed a phase transition could automatically be banished to the class called NP-complete. But the situation has turned out to be more subtle. Some easier problems like the 2-SAT variety also undergo phase transitions. But it's recently become clearer that for 2-SAT problems the transition from soluble to insoluble is relatively smooth, while with 3-SAT problems it is abrupt.

The study published last week goes further. After studying random pools of formulas that mix 2- and 3-actor clauses, the authors conjecture that, in general, simple P problems will have smooth transitions, while NP-complete problems will have abrupt transitions. If this can be more rigorously shown, a breakthrough may be at hand.

Most interesting of all, the authors suggest why the abrupt transitions occur. Scientists often study complex systems by comparing them with an idealized substance (simulated on a computer) called a spin glass. These alloys contain a random distribution of magnetic particles that, roughly speaking, can point up or down. Think of each particle as one of the actors whose demands must be met. "Up" means in the play, "down" means out. Just as the presence or absence of an actor can affect whether others remain in the cast, the orientation of a magnetic particle can affect the way its neighbors line up. Solving a satisfiability problem becomes equivalent to getting the spin glass to settle down into a state with all the particles comfortably coexisting.

The authors suggest that whether one is looking at SAT problems or spin glasses, abrupt phase transitions are caused by the formation of a "backbone" of particles (or actors) that are permanently frozen into one state or the other (like Miss Alvarez refusing to act with Mr. Cohen). Understanding the formation of these structures could lead to new problem-solving techniques for both physicists and computer scientists.

In the past, the flow of information between these two fields has been largely one-way, to the benefit of the computer scientists. Dr. Selman hopes some of his group's insights will flow back the other way, deepening the understanding of the physical world. "It's a sign," he said, "of computer science becoming more mature."