UNCOMMON FACTORING

New computing machines and new algorithms are speeding up the factoring of large numbers

By IVARS PETERSON

It's easy to see that $10^{1031}-1$ is not a prime number. Written out in full, the number consists of 1,031 nines, into which, of course, nine divides evenly. What's left is a string of 1,031 ones. Is this new number divisible only by one and itself (the definition of a prime number), or is it also the product of two or more primes? This question turns out to be much harder to answer.

Mathematician Hugh C. Williams of the University of Manitoba in Winnipeg suspects that it is a prime number, and for months he has been collecting clues to settle the question. He now may have enough information to come to a conclusion. "It's simply a matter of putting it together," he says.

Williams has been helped by recent, rapid advances in methods for factoring large numbers. Much of this work has been pushed ahead because of interest in the security of cryptosystems based on the difficulty of factoring. Mathematicians are coming up with new factoring algorithms, and new machines specially designed for factoring are starting to appear.

Only 10 years ago, just a handful of number theorists cared about factoring numbers. "We lived in obscurity," says Williams. Then, computers went forth and multiplied, and concurrent developments in cryptography showed that these problems had practical value. This brought many more people into the field. As a result, in less than four years, successful factorizations of "hard" numbers jumped from 50 to 71 digits. Every number on a famous list of the 10 "most wanted" factorizations has also been factored.

At the Sandia National Laboratories in Albuquerque, N.M., Gustavus J. Simmons, James A. Davis and Diane B. Holdridge have been busy fine-tuning a factoring method called the "quadratic sieve" (QS) so that it runs efficiently on a Cray X-MP supercomputer. A year ago, the Sandia team set a record for the largest "hard" number ever factored by a general-purpose factoring method (SN: 1/14/84, p. 20; 3/17/84, p. 171). This 71-digit number was cracked in 9.5 hours of computing time.

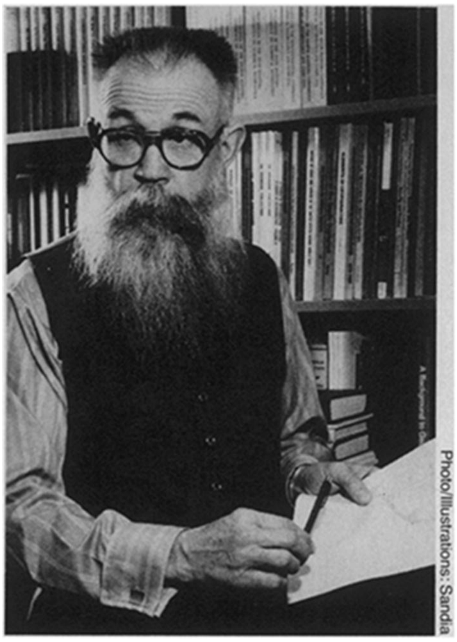
"For the past year, we've been working very hard on improving our code and adapting the program to multiprocessor machines," says Simmons. Now, "one day's running on a Cray X-MP, with both of the computer's processors available, will factor a 77- or 78-digit number of the same difficulty as the 71-digit number."

Although the researchers haven't tried such a number yet, Simmons has no doubts about being able to do it. However, "we're not a factoring factory," he says. "Our business, because we're concerned with cryptography, is to provide the sharpest estimates ... as to how difficult factoring is. To spend a day's time factoring a 77-digit number doesn't provide as much information as taking a 71-digit number that has been previously factored and then doing it again with a new algorithm."

On the horizon is a new set of supercomputers, the Cray-2 and its Japanese competitors. These machines will probably run the Sandia factoring program about 10 times faster, predicts Simmons. "This new class of machines, with the present state of the art in mathematics, means factoring numbers of better than 85 digits in a day's time."

Researchers at the University of Georgia in Athens are taking a different approach. There, computer scientist Jeffrey Smith, instead of tinkering with an algorithm to match it to a particular computer, is building a computer to suit the algorithm. His machine is specifically designed to run the "continued fraction" factoring method. This computer, called an "Extended Precision Operand Computer" or EPOC, is also known colloquially as the "Georgia Cracker."

"The EPOC is a 'let's see what we can do' type of thing," says Smith. The main part of the machine is now running, and it has already factored some numbers with up to 56 digits. The EPOC is slower at factoring than a Cray supercomputer, but it is also much cheaper. Costing only about $10,000 in parts to build, the machine can run as long as necessary to come up with an answer. While the EPOC may take two weeks to factor a 71-digit number, valuable



Photo/Illustrations: Sandia

Sandia's Gustavus J. Simmons points out improvements in factoring.

supercomputer time isn't used up doing the problem.

"In principle, we can factor a number of any length," says Smith. "It's just that the universe may grow cold by the time we get an answer."

"They won't set any records with it," says Simmons. "But they will do a great deal of important factoring. It will make production factoring very economical."

Smith is now designing a new, more powerful machine that will fit the "quadratic sieve" algorithm, invented by Georgia's Carl Pomerance and being used in a modified form at Sandia. "If we can show that we can build special-purpose processors within a reasonable time and make them successful," says Smith, "then that will really give people a lesson about the ease of building their own processors."

Marvin C. Wunderlich, now at the National Security Agency (NSA) in Fort Meade, Md., is also working with the "continued fraction" factoring algorithm, but he is using a "Massively Parallel Processor," or MPP. This computer, originally built for the National Aeronautics and Space Administration to do image processing, contains 16,384 small processors or "bit-pushers," which can perform, in step and at the same time, a large number of simple computations.

The trick is to modify the algorithm so that it takes advantage of the parallel processing that the MPP allows. In addition, Wunderlich is trying to implement an "early abort" feature. "This is just a means of cutting your losses early," says Williams, who recently joined Wunderlich to help with solving some of the problems in adapting the algorithm to the computer. "If you see that something isn't working, get rid of it instead of continuing to work with it." These improvements could make the "continued fraction" method competitive with the Sandia group's "quadratic sieve."

When it starts factoring numbers this spring, the MPP will probably be able to factor 60-digit numbers faster than the Cray, says Simmons. But it loses this advantage beyond 75 digits or so. "If someone like NSA wanted to build a machine with even more parallelism, going up to maybe 128,000 parallel channels," he says, "then it would move up to where we are now on the Cray." Wunderlich's work, however, will provide a benchmark for the speed that can be obtained at a given level of parallelism.

"No one really knows how hard factoring is," says Williams. "My interest is to see how well the algorithm can function under conditions that should be optimal for its running. In attempting to implement the algorithm, we may also learn something about it, and this could permit us to make it go faster."

Factoring at Sandia: Doing the 10 "most wanted" factorizations

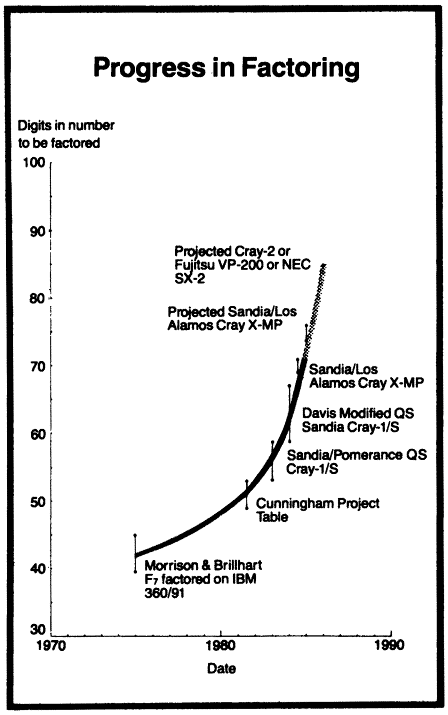
Number	No. of digits in "hard" part	No. hours to factor
2^{211}-1	60	22.25
2^{251}-1	69	32.3
2^{212}+1	54	1.0
10^{64}+1	55	4.4
10^{67}-1	61	1.22
10^{71}-1	71	9.5
3^{124}+1	58	1.8
3^{128}+1	53	6.05
11^{64}+1	67	15.34
5^{79}-1	55	0.99

Sandia mathematicians have now factored all 10 numbers on a special list of numbers that were of particular interest in mathematics research.

Duncan A. Buell and his colleagues at Louisiana State University (LSU) in Baton Rouge are also custom building a special computer. This one is designed to handle calculations involving a large number of decimal places.

Computers normally handle instructions and computations as "words" — packages of bits — with a fixed length. Most personal computers, for example, use 8-bit or 16-bit words. However, for calculations that require answers with, say, 75 or 100 decimal places, these words must be strung together. It takes a lot of programming to make those strings behave like single numbers.

The LSU computer will have 256-bit words that can be broken up into eight 32-bit slices. These slices can be connected or disconnected at will. "If we need 128-bit multiplies, we can do one at a time," says Buell. "If we need 32-bit multiplies, we can do four of these at once." A 256-bit machine is large enough to allow the factoring of a 76-digit number, he says.



"We hope to have a general-purpose machine that can be programmed so that algorithms can take advantage of the hardware," says Buell. "We think we can get extremely fast processing for certain kinds of factoring algorithms."

Until recently, Buell was looking at a factoring algorithm invented by Hendrik W. Lenstra Jr. of the University of Amsterdam in the Netherlands and Claus P. Schnorr of the University of Frankfurt in West Germany. Because this factoring method involved the use of numbers accurate to a large number of decimal places, it had not been widely used. Earlier this year, however, Lenstra came up with a faster, simpler version of the algorithm (SN: 3/9/85, p. 151).

"The new Lenstra method requires substantially less overhead in doing the computations than does the original method," says Buell. "Our machine should work on this new algorithm very well. I think we have the right kind of processors in the right kind of arrangements."

Curiously, the five or six best general-purpose methods available for factoring seem to share at least one feature. As the result of refinements in the last few months, all of them now have approximately the same upper limit on their running times. Although such limits are not mathematically rigorous, they are considered to be reasonable estimates of how long a particular algorithm would take to do its job.

"Are we really seeing the true level of difficulty of factoring integers?" asks Andrew M. Odlyzko of AT&T Bell Laboratories in Murray Hill, N.J. "Or is it that we are still blind, that we still can't see something?"

"At the moment, no one can give any sort of reading on just what this means," says Williams. "A brand new idea would be one that gets beyond that particular asymptotic limit." He adds, "I really don't know what the future holds, but I'm very optimistic. I think that we may see some wonderful things."

Meanwhile, at odd moments, Williams ponders his string of 1,031 ones. Last year, in his article "Factoring on a Computer" in THE MATHEMATICAL INTELLIGENCER (Vol. 6, No. 3), he challenged readers to find all the factors of the number $10^{103}+1$, which would help him prove that $(10^{1031}-1)/9$ is a prime. He already knew some of the factors: 11, 1,237, 44,092,859, 102,860,539 and 984,385,009. Dividing the original number by all these known factors left a 75-digit number that was tough to crack.

Responding to the challenge, two mathematicians, using a special-purpose factoring method that required 220 minutes of computer time, finally found the remaining three prime factors. Now, the answer to Williams's original question is practically within reach. □