Computation

Highly distributed processing

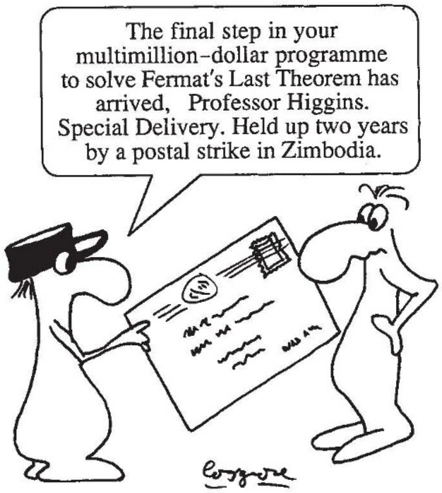
Ian Stewart

What is the best way to obtain raw computational power? A massive central supercomputer or a network of smaller machines? Both have their passionate devotees. A recent calculation in number theory has given a new meaning to the phrase 'distributed processing': it illustrates the possible advantages of a network for problems that can be broken into independent pieces. A 100-digit number, carefully screened for maximum difficulty, has been resolved into prime factors by the communal efforts of a dozen mathematicians in three countries using 400 separate computers simultaneously and communicating by electronic mail. The marathon calculation widely reported in the popular press (see *Nature* **335**, 658; 1988) was organized by Mark S. Manasse of the Digital Equipment Corporation's research centre at Palo Alto and Arjen K. Lenstra of the University of Chicago. It took less than a month: a supercomputer working non-stop would have taken at least a week and cost a great deal more.

The factorization of large numbers into primes used to be of purely theoretical interest. Today, however, the problem has important applications in cryptography and cryptanalysis. A well known method of encoding sensitive information, the RSA (Rivest–Shamir–Adelman) system, relies for its security on the conjectured difficulty of resolving an arbitrary number into its constituent primes. Any improvement in factorization methods is thus automatically of interest in a much wider sphere. The communal effort under discussion is in a sense most noteworthy for its *lack* of novelty. The actual method employed, known as the quadratic sieve, was invented by Carl Pomerance in 1982 (*Computational Methods in Number Theory Part I* (eds Lenstra, H. W. Jr & Tijdeman, R.) Mathematisch Centrum Tract **154**, Amsterdam; 1982), and was rapidly accepted as one of the fastest methods available.

The first thing to understand about prime factorization is that the most obvious method, trial division of the number N by each possible prime in turn up to the square root, is hopelessly inefficient for numbers with more than just a few digits. The number of primes less than $\sqrt{N}$ is approximately $2\sqrt{N}/\log N$, so the number of trials needed for a 100-digit number is roughly $2 \times 10^{50}/(50\log 10) = 1.7 \times 10^{48}$. At a million trials per second the computation would take about 5×10^{34} years, or at least 10^{27} times the age of the Universe. Variations on this technique, such as strategies for searching specific ranges of primes, like starting at $\sqrt{N}$ and working backwards, suffer from the same defect.

The solution is to apply more subtle methods, based on results in number theory. Suppose we wish to find the prime factors of a number N. If we can find just one of them, we can divide it out and repeat the process on the much smaller number that results. So the real problem is to find *some* prime factor. One approach is to look for two perfect squares whose difference is a multiple of N. Let x^2 and y^2 be these squares. Because $x^2-y^2 = (x-y)(x+y)$, any prime factor of N must divide either $x-y$ or $x+y$. Suppose for the sake of argument that it divides $x+y$; then it also divides the highest common factor of $x+y$ and N. There is a very efficient method, known as the euclidean algorithm, for computing this highest common factor, which in general is much smaller than N. Because it is smaller, we can find its prime factors relatively easily; but these are also prime factors of N. The strategy is thus very much one of 'divide and conquer'.


The final step in your multimillion-dollar programme to solve Fermat's Last Theorem has arrived, Professor Higgins. Special Delivery. Held up two years by a postal strike in Zimbodia.

It's a neat trick — but how do we find x and y? Michael Morrison and John Brillhart (*Mathematics of Computation* **29**, 183–205; 1975) realized that suitable candidates for x and y could be found by using the continued fraction for $\sqrt{N}$, a sequence of fractions p/q that approximate $\sqrt{N}$ ever more closely. Their proposal was to run along this sequence of fractions waiting for the denominator q to become a perfect square. When this happens, suitable x and y can easily be found. But there is a much better way. Instead of waiting for a square denominator to turn up, we can try to create one. Some upper limit is chosen and each denominator is checked to see whether all of its prime factors lie below this limit. If so, its factorization is stored as a sequence of zeros and ones: 0 if the corresponding prime factor occurs to an even power, 1 if it occurs to an odd power. In this way a huge matrix is built up, with these sequences as its rows.

Some subset of all these denominators, multiplied together, will form a square — provided that the corresponding sequences have, between them, an even number of ones in each column. This can be checked quickly by standard techniques of linear algebra, known as gaussian elimination. Having found a combination of denominators whose product is a perfect square, numbers x and y such that N divides x^2-y^2 are again easy to find. This is the basic idea: the full-blooded quadratic sieve incorporates several further refinements, including strategies to cut short the computation if it is getting nowhere. (For detailed description see *Prime Numbers and Computer Methods for Factorization* by Hans Riesel; Birkhäuser, Boston, 1985).

Manasse and Lenstra's collaborators were from the United States, the Netherlands and Australia. The matrix required to apply the quadratic sieve turned out to have 50,000 rows and columns, a total of 250 million entries. The computation of the matrix was split into 400 pieces, the results were assembled by electronic mail at Palo Alto, and the final gaussian elimination led to the discovery that the number had two prime factors, with 40 and 61 digits. There are so many computers now that much of their time is effectively free, especially for low-priority jobs running 'in background', and the tale bears dramatic witness to the capabilities of a network of small machines running in parallel.

The quadratic sieve is a highly refined and extremely clever technique, but has by now become fairly standard. So in a sense the method is an example of brawn rather than brains. The particular style of brawn, large-scale international cooperation, is seldom applied in such an organized manner. On the other hand, the whole of mathematics is a large-scale international cooperation, whose customary methods of communication are often a good deal slower than electronic mail.

By pushing the present approach to its limits, factorizations of 110- or perhaps 120-digit numbers might be achieved. To factorize an arbitrary 150-digit number, say, still looks out of the question, and the only hope is a completely new idea. This is not inconceivable; for example Hendrik Lenstra (of Amsterdam University) recently suggested a new approach involving so-called elliptic curves (see my News and Views article in *Nature* **325**, 199; 1987), which can sometimes be extremely efficient. But no known method works efficiently for the awkward cases as well as the nice ones. In the world of prime factors, nobody yet knows whether brain will ultimately prove superior to brawn. □

Ian Stewart is in the Mathematics Institute, University of Warwick, Coventry CV4 7AL, UK.