Reference: Christoph Pöppe, "Der Computer als Formelentdecker", *Spektrum der Wissenschaft*, Jan 2009, pg. 76-78. Translation below by Roland Girgensohn and David H. Bailey.

The Computer as a Discoverer of Formulas

A computer program can discover through purposeful trials what a numerically calculated number "really" is. But this will not put mathematicians out of employment. On the contrary, using the program, they find numerous relationships that then need a proof.

By Christoph Pöppe

While investigating an interesting geometric object and trying to determine its shape in different planes, I use the theorem on intersecting lines in one plane, invoke Pythagoras in another plane, solve a few equations, put the results into other equations... After a lot of computing and after correcting the most obvious errors, I obtain, for the quantity that interests me, a very ugly expression, with roots nestled within roots and worse. What can I do with it?

I put the expression into a computer algebra system such as "Maple" or "Mathematica" and check what the built-in simplification procedures make of it. They easily find common denominators for even the most unwieldy fractions, expand brackets and tame the flood of the resulting terms, even if there are lots of unknowns. But if it's really roots within roots, they often do not find anything to simplify.

In this case it is worthwhile to simply evaluate the hard-won formula numerically – plugging in values for the unknowns if the formula contains any. Suddenly there appears something like 0.866 – and I recognize this number at first glance as an old acquaintance. It is $\sqrt{3} / 2$, the ratio of height to side length in an equilateral triangle. With this newly acquired insight, I take a look at my geometric construction again and, lo and behold: here is the equilateral triangle from which everything I need can easily be deduced! I needed just this one friendly hint about the numerical value to now be able to find a rigorous proof.

Infinite sums are similarly amenable to this kind of "trial-and-error computing". What is the value of $-1 / 3 + 2 / 9 + 3 / 27 + 4 / 81 + ... + n\,(-1/3)^n$ ...? [these terms should have alternating signs – DHB]. If one adds the first 20 terms one obtains the value -0.1875. This does not noticeably change when taking 40 instead of 20 terms, therefore one keenly concludes that the sum will probably be -3/16. And after some searching about in references, one finds for this kind of sums a closed-form representation, which confirms the result obtained through trial and error.

That is all very nice and may sometimes provide an original approach to a long-known result – but can you really find something new by these primitive means? Imagine that Leonhard Euler (1707 - 1783), arguably the most productive and ingenious mathematician of all time, had had access to a modern computer. He would certainly have known how to use it, for example, to compute the first 10,000 terms of his famous sum $1 + 1/4 + 1/9 + ... + 1/n^2 + ...$ and thus to at least approximately determine the value of the series. It turns out that this tedious work, which with paper and pencil even Euler probably would not have taken upon himself, results in just three correct decimal digits. And from this meager information one should infer that the result is π^2? That would have been asking too much of even the master of computation.

Euler actually did it quite differently (see *Spektrum der Wissenschaft Spezial*, 2005, issue 2, "Infinity (plus one)," p. 19). He takes a function that bears no discernible relationship to the sum over $1/n^2$ or to the constant π^2, namely the sine function, looks at it from two different perspectives, infers an algebraic relationship from the fact that both images nevertheless show the same object – and from the thicket of derivations Euler's famous formula jumps like a rabbit out of the hat.

For all those who do not share Euler's genius, there is now good news: trial-and-error works beyond these simple examples. You will celebrate feasts of recognition, without having to memorize the digits of π or $\sqrt{3}$, let alone those of $\sqrt{3}/2$ or $\pi^2/6$.

Scientific trial-and-error

There is an algorithm which takes as input a number, expressed as a decimal fraction. This number may be the value of a finite sum as an approximation to an infinite one or it may be the result of some opaque calculation. Or it may be the solution of an equation, found numerically, i.e., by a step-by-step improvement of some approximation. We may suspect that this number can be expressed by, say, π, $\sqrt{2}$, $\sqrt{5}$ and perhaps the natural logarithm of 2, and that the expression contains, apart from these non-smooth numbers, only integers. Then the algorithm will either produce these integers, or it will report that if they exist at all, they must be of astronomical magnitude.

Actually, one would prefer the clear-cut information "these numbers exist" or "they don't exist." But there is no algorithm that can accomplish this, for fundamental reasons: numerical calculations always have only finite accuracy. In standard computer applications, this is about 16 decimal places. With this accuracy one can represent any number as, for example, $a_1 \pi + a_2 \sqrt{2}$, where a_1 and a_2 are integers; but generally a_1 and a_2 will then themselves have a magnitude of 10^{16} [actually about 10^8 – DHB]. Therefore only relations with small integers are interesting.

Following several predecessors, an algorithm called "PSLQ" is now the state of the art for finding these integer relations. Its author, Helaman Ferguson, is less known to the general public for this achievement – even though PSLQ is listed as one of the ten most important algorithms of the [20th] century – but rather for his mathematical sculptures. He carves objects such as three-sided tori from stone, and his bronze sculpture "Alexander's Horned Sphere" has achieved cult status.

Still, there is one parallel between PSLQ and the work of a stone mason: Both really require a lot of strength and patience. The computation times become bearable only with a 2004-era PC or higher.

It should be noted that although PSLQ can find such identities with integer coefficients, this does not constitute a proof. There are enough deterrent examples of “false friends,” which look like true identities, but aren’t. In 1975, Martin Gardner stated in his column “Mathematical Games” in *Scientific American* that $e^{\pi \sqrt{163}}$ was an integer – which was an April Fool’s joke. The assertion turns out to be wrong only in the twelfth digit after the decimal point. But doubts about an identity rapidly vanish if it turns out to hold up to over 100 or 200 decimal places.

Moreover, the effectiveness of the PSLQ algorithm improves when the input is provided to very high precision. Therefore it only really achieved eminence when David Bailey of Lawrence Berkeley National Laboratory at Berkeley (California) coupled it with his high-precision arithmetic, a set of computer programs that can add and multiply numbers with several thousand digits after the decimal point. Jonathan M. Borwein of Dalhousie University in Halifax (Canada) and other mathematicians teamed up with Bailey and have used the new procedure to reap a rich harvest of results.

The first spectacular result of their efforts is the “cable car for π” (*Spektrum der Wissenschaft* 5 / 1997, p. 10): You do not have to calculate the digits of π one by one as usual, but you can start directly with the millionth and following digits, without having to know the preceding ones. Provided, however, that you do not use decimal digits. The cable car only works if you use digits to the base 16 (or base 2, which amounts to the same thing, since 16 is a power of 2).

In this case, even the above-mentioned usual weakness of numerical computing, namely limited accuracy, does not spoil the fun. The hexadecimal digits of π are integers between 0 and 15. In that case it is sufficient to know that the unknown is equal to 5 with an error of 10 percent, because then it cannot be anything other than a 5 [“10 percent” here should be “6 percent” or smaller – DHB].

True computer proofs

In this and similar cases, the search procedure PSLQ therefore not only produces well-founded conjectures, for which it may then be worthwhile to look for a proof, but actually full-blown proofs [this is not entirely correct – RG]. These proofs are comparable to the proofs of the four-color theorem or of Kepler’s conjecture (*Spektrum der Wissenschaft* 4 / 1999, p. 10): They just use the computer for computations that a human, with his head and in his lifetime, could not manage without mistakes. There remains, however, especially among traditionally-minded mathematicians, the unsatisfactory feeling that an essential part of the idea of the proof is missed.

By now, the research area has grown so much that its representatives have proudly declared it a new field: “Experimental Mathematics,” complete with introductory courses, scientific meetings, a trade journal – and a public call to the masters of the field to show their skills. In this, the experimental mathematicians followed the lead of their colleagues from classical numerical analysis. In 2002, Nick Trefethen, from the University of Oxford, published a collection of ten problems under the title "100 Digit Challenge", which was meant to be a hard nut to crack for his colleagues from numerical analysis. This was followed by Bailey and his colleagues with ten problems in their field.

The assortment of goods is truly varied. One problem arises from chaos theory: to obtain an algebraic equation for the point at which the stable solution of a dynamic system of period 4 changes to period 8. Or you have to calculate the force on an ion inside an infinitely large salt crystal. Or to investigate the properties of a sequence which in turn is related to certain continued fraction expansions (picture p. 76/77).

Or to shed light on a random discovery: In 1993, the student Enrico Au-Yeung computed the first 500,000 terms of the infinite sum

$$1 + (1 + 1/2)^2/4 + (1 + 1/2 + 1/3)^2/9 + ... + (1 + 1/2 + ... 1/k)^2/k^2 + ...$$

and obtained the result 4.59987...; for these six digits this is exactly equal to $17\,\pi^4/360$. He brought this to Jonathan Borwein's, his supervisor's, attention, who at first regarded this as an insignificant coincidence, comparable to Gardner's "false friend". But he anyhow started tossing the series about, until he transformed the sum to an integral. The integral did not really look nicer, but it could be computed with just moderate effort to considerably more digits. To his surprise the two values now agreed to over 30 digits!

General attack on $17\,\pi^4/360$

Now of course Borwein wanted certainty. He mustered the whole arsenal of analysis, including the notorious zeta function, the subject of the Riemann hypothesis (*Spektrum der Wissenschaft* 9 / 2008, p. 86). And even more: Borwein generalized the zeta function to several variables, defined a new quantity that provides a relation between several of such generalized zeta functions, and used PSLQ to find an integer relation for it - and at the end solved by a mixture of algebraic and numerical computing a whole class of problems in which the original is only a special case.

Of course, the success of this whole enterprise crucially depends on the successful outcome of the computer-based search for an integer relation. But it would be completely untrue to say that this reduces the work of mathematicians to the mindless usage of a software package. A great deal of creativity is already required just to prepare the problem in such a way that it becomes palatable for the software. In addition, the computer algebra systems which serve for such work as a sort of programming language have become so rich and complicated that they have developed something like personal quirks. While they can't make head or tail of a problem in one form, they brilliantly solve it in another form, and nobody knows why.

Experimental Mathematics is a unique blend of computing – both numerically as well as with algebraic symbols – and classical mathematical methods.

Christoph Pöppe is an editor for *Spektrum der Wissenschaft*.

Bailey, D. H. et al.: *Experimental Mathematics in Action*. A K Peters, Wellesley (Massachusetts) 2007.

Bailey, D.H. et al.: “Ten Problems in Experimental Mathematics.” In: *The American Mathematical Monthly* 113, p. 481 - 509, 2006.

Bailey, D.H., Borwein, J.M.: “Experimental Mathematics: Examples, Methods and Implications.” In: *Notices of the American Mathematical Society* 52, p. 502 - 514, 2005.

Bornemann, F. et al.: *The SIAM 100-Digit Challenge: A Study in High-Accuracy Numerical Computing*. SIAM, 2004.

[Sidebar on page 78:]

Let $x_1, x_2, ..., x_n$ be n real numbers, given with a potentially large but finite decimal accuracy. The problem is to find integers $a_1, a_2, ...,, a_n$, so that $a_1 x_1 + a_2 x_2 + ... + a_n x_n = 0$. The “trivial” solution of the problem – to set all a’s equal to 0 – is expressly excluded.

Many problems of experimental mathematics can be formulated in this way. The two most common examples are:

* You have a numerically calculated number y, and there are good reasons for believing that it is a rational linear combination of certain constants such as π, π^2, log 2 and so on. That is, with some rational numbers r_1, r_2, r_3 one can write $y = r_1 \pi + r_2 \pi^2 + r_3 \log 2$. Multiply this equation by the common denominator of r_1, r_2 and r_3, bring everything to the left-hand side of the equation, and you will have reached the above standard form of the problem.

* You want to know whether a numerically calculated number x is algebraic, or in other words, whether integers $a_0, a_1, a_2, ..., a_n$ exist such that $a_0 + a_1 x + a_2 x^2 + ... + a_n x^n = 0$. Then it is sufficient to let $x_1, x_2, ..., x_n$ be the numbers $1, x, x^2, x^3, ...$

Geometrically speaking, the problem consists in finding in an n-dimensional vector space a vector $(a_1, a_2, ...,, a_n)$ which is orthogonal to the given vector $(x_1, x_2, ..., x_n)$, but has only integer (and as small as possible) components. Even if you look for such integer vectors only up to a fixed bound, then although there are only finitely many possible vectors, it will still be hopeless to try them all.

Instead, the PSLQ method converts this problem first into a matrix eigenvalue problem. For such problems there are highly sophisticated solution procedures. Helaman Ferguson has worked out one of these procedures such that it is suitable for high precision arithmetic and does not suffer from the numerical instabilities that plagued its predecessors.