Introduction to Experimental Mathematics

Victor Adamchik

Carnegie Mellon University

Integer Relation Algorithms

"Computers are useless. They can only give you answers."

Pablo Picasso

"The purpose of computing is insight, not numbers."

Richard Humming

Given a vector of real number $\{x_1, \ldots, x_n\}$, find a vector of integers $\{p_1, \ldots, p_n\}$ such that a linear combination of given numbers is zero, namely

$$p_1 x_1 + \ldots + p_n x_n = 0$$

The algorithm was discovered iun 1979 by Ferguson and Forcade [1].

In 1982 it was improved by Lenstra, Lenstra, Lovász [2].

1992, more improvements by Ferguson and Bailey - PSLQ algorithm[3].

2D case:

Suppose there are two numbers x, y. Find integers n, m such that

$$x\,n + y\,m = 0.$$

If x and b are integers, we use the **Euclidean** algorithm

$$\begin{array}{ccccc} x & = & y * q_1 + r_1, & & 0 \le r_1 < y \\ y & = & r_1 * q_2 + r_2, & & 0 \le r_2 < r_1 \\ r_1 & = & r_2 * q_3 + r_3, & & 0 \le r_3 < r_2 \\ & \ldots & \ldots & & \ldots \\ r_{k-2} & = & r_{k-1} * q_k + r_k, & & 0 \le r_k < r_{k-1} \\ r_{k-1} & = & r_k * q_{k+1} + 0 & & \end{array}$$

What if we apply this idea to real numbers?

$$\text{GCD}\left(\sqrt{2},\ 1\right)$$

1.414214 = 1 * 1 + 0.414214

1 = 2 * 0.414214 + 0.171573

$0.414214 = 2 * 0.171573 + 0.071068$

$0.171573 = 2 * 0.071068 + 0.029437$

$0.071068 = 2 * 0.029437 + 0.012193$

and so on

Since remainders $r_k \to 0$ on each iteration, we will get either an exact relation or an approximation.

This is a cornerstone idea of the lattice reduction algorithm.

Infinite continued fraction for $\sqrt{2}$:

$$\sqrt{2} = 1 + \cfrac{1}{2 + \cfrac{1}{2 + \cfrac{1}{2 + \cfrac{1}{2 + \cfrac{1}{2 + \ldots}}}}}$$

Lattice Reduction Algorithm (LLL algorithm)

Let B be a set of vectors B = $\{\boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_m}\}$ in $\mathbb{Q}^n$. If they are independent then they form a basis, which means that any point can be written as a linear combination of $\boldsymbol{b_i}$

$$x = \sum_{k=0}^{m} r_i \, \boldsymbol{b}_i$$

here coefficients r_i are real numbers. Now instead of real r_i we choose only integers

$$L = \left\{ \sum_{k=0}^{m} n_i \, \boldsymbol{b}_i, \quad n_i \text{ is an integer} \right\} \subseteq \mathbb{Q}^n$$

The set of such points forms a lattice L. This lattice has dimension n and rank m.

Suppose we have two vectors

$$b_1 = \{1, 0\}$$
$$b_2 = \{0, 1\}$$

What is the lattice formed by these vectors?

Now given a lattice, the basis B of course is not unique, and we may look for bases with some *distinguished* properties. We would like to reduce B to basis B', also describing L, where B' is a "good" lattice basis in the sense of some reduction theory - the basis which has a shortest vector.

The Euclidean length of a vector $V = \{v_i, v_2, \ldots, v_k\}$ is defined by

$$|V| = \sqrt{\sum_{i=1}^{k} v_i^2}$$

■ Example

Suppose we have two vectors

$$b_1 = \{1,\ 9\}$$
$$b_2 = \{4,\ 37\}$$

The length of each vector is

$$\sqrt{1^2 + 9^2} = 9.05\ldots$$

$$\sqrt{4^2 + 37^2} = 37.21\ldots$$

We can reduce the basis by the following transformations

$$b_2 = b_2 - 4\ b_1;$$

$$b_1 = b_1 - 9\ b_2;$$

```
b1 = {1, 9}; b2 = {4, 37};
b2 = b2 - 4 b1;
b1 = b1 - 9 b2;
{b1, b2}
```

```
{{1, 0}, {0, 1}}
```

The new basis is shorter comparing to the original - each length is just 1.

In *Mathematica* the basis reduction is done by LatticeReduce:

```
LatticeReduce[{{1, 9}, {4, 37}}]
```

```
{{0, 1}, {1, 0}}
```

The problem of finding the shortest vector is believed to be NP-complete [4].

However, an approximate solution algorithm [2] - known as the LLL, runs in polynomial time

Why would be we interested in a shortest vector? Consider the following basis vectors

$$1, 0, 0, \ \ldots, 0, C*\tau_1$$
$$0, 1, 0, \ \ldots, 0, C*\tau_2$$

$$0,\ 0,\ 1,\ \ldots,\ 0,\ C*\tau_3$$
$$\ldots$$
$$0,\ 0,\ 0,\ \ldots,\ 1,\ C*\tau_n$$

where C is a constant (usually, huge) and τ_i are rational approximations of the real numbers x_i. Now suppose we are able to reduce this basis to a "good" one, the basis which has a short Euclidian length. Each vector w of the new basis will look like

$$w = \left\{w_1,\ w_2,\ \ldots,\ w_n,\ C*\sum_{i=i}^{n} w_i\,\tau_i\right\}$$

If this is a shortest vector then

$$C*\sum_{i=i}^{n} w_i\,\tau_i\ \to\ 0$$

is small or maybe zero. This means that if we replace approximations τ_i by real numbers

$$\sum_{i=i}^{n} w_i\,x_i = 0$$

we get a new identity for $x_1,\ x_2,\ \ldots,\ x_n$. For better understanding, let us consider a few examples from [5, 6, 7].

■ **Example** (finding minimal polynomials)

Given a real algebraic number α = 1.3027756377319946465596. *Find the minimal polynomial for it.*

If α is algebraic then there is such integer p that

$$\{1,\ \alpha,\ \alpha^2,\ \ldots\,\alpha^p\}$$

has an integer relation. We start with the basis

```
B := {{1, 0, 0, 0, 0, c },
    {0, 1, 0, 0, 0, c α},
    {0, 0, 1, 0, 0, c α²},
    {0, 0, 0, 1, 0, c α³},
    {0, 0, 0, 0, 1, c α⁴}};
```

where arbitrary constant c is chosen to be 10^{15} - the bigger the better.

```
α = 1.302775637731994646559 6;
c = 10^15;
B // MatrixForm
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1\,000\,000\,000\,000\,000 \\ 0 & 1 & 0 & 0 & 0 & 1.3027756377319946465596\times 10^{15} \\ 0 & 0 & 1 & 0 & 0 & 1.6972243622680053553440\times 10^{15} \\ 0 & 0 & 0 & 1 & 0 & 2.2111025509279785862 38\times 10^{15} \\ 0 & 0 & 0 & 0 & 1 & 2.8805705358760374740 83\times 10^{15} \end{pmatrix}$$

Next, we reduce this basis

```
Round[N[B, 30]];
LatticeReduce[%];
N[%]
```

$\{\{-3., 1., 1., 0., 0., -2.85695\times 10^{-7}\}, \{0., -3., 1., 1., 0., -7.60105\times 10^{-9}\}, \{0., 0., -3., 1., 1., 1.51839\times 10^{-6}\}, \{-581\,543., -1.68918\times 10^{6}, -55\,447., -5.0121\times 10^{6}, 4.84575\times 10^{6}, 4.9881\times 10^{6}\}, \{757\,621., 2.20063\times 10^{6}, 72\,240., 6.52964\times 10^{6}, -6.31293\times 10^{6}, 1.14865\times 10^{7}\}\}$

Among new vectors, we need to pick the shortest one

```
%[[2]] .{1 , x , x^2 , x^3 , x^4, 0}
```

$0. - 3.\,x + 1.\,x^2 + 1.\,x^3 + 0.\,x^4$

Therefore, we conject that the minimal polynomial for number α is

$$x^2 + x - 3$$

■ **Example** (trigonometry)

Using LLL algorithm, find unknown coefficients r_1 and r_2:

$$\cot\left(\frac{\pi}{8}\right) + \cot\left(\frac{2\pi}{8}\right) + \cot\left(\frac{3\pi}{8}\right) \to r_1 + r_2\sqrt{2}$$

We start with the basis

```
B := {{1, 0, 0, c 1}, {0, 1, 0, c √2}, {0, 0, 1, c V}};
B // MatrixForm
```

where V is a numeric approximation of

```
V = N[Cot[Pi / 8] + Cot[Pi / 4] + Cot[3 Pi / 8], 40];
```

Next, we reduce this basis

```
Round[N[B, 30]];
LatticeReduce[%] // N
```

```
{{-1., -2., 1., 4.76464 × 10^-15},
 {2.41811 × 10^7, -1.28713 × 10^7, -1.56155 × 10^6, 2.21048 × 10^7},
 {-3.41972 × 10^7, 1.82028 × 10^7, 2.20837 × 10^6, 3.12609 × 10^7}}
```

Among new vectors, we need to pick the shortest one, this is the first one in a list. This yields

```
Clear[V];
Rationalize[%%[[1]]] .{1 , √2 , V, 0}
```

```
-1 - 2 √2 + V
```

$$-1 - 2\sqrt{2} + V == 0$$

■ **Example** (integration)

$$\int_0^\infty \frac{\sqrt{x}\,\log^2(x)}{(1-x)^2}\,dx = 2\pi^2$$

$$\int_0^\infty \frac{\sqrt{x}\,\log^3(x)}{(1-x)^3}\,dx = -3\pi^2 + \frac{\pi^4}{4}$$

The question: what is

$$\int_0^\infty \frac{\sqrt{x}\,\log^4(x)}{(1-x)^4}\,dx = ??$$

Looking at two previous results we guess that the integral is a linear combination of π in even powers:

$$r_1 + r_2\pi^2 + r_3\pi^4 + r_4\pi^6$$

where coefficients r_i are unknown. We find them using the LLL algorithm.

Start with the basis

```
B := {{1, 0, 0, 0, 0, c 1}, {0, 1, 0, 0, 0, c π^2}, {0, 0, 1, 0, 0, c π^4},
   {0, 0, 0, 1, 0, c π^6}, {0, 0, 0, 0, 1, c V}};
B // MatrixForm
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1\,000\,000\,000\,000\,000 \\ 0 & 1 & 0 & 0 & 0 & 1\,000\,000\,000\,000\,000\,\pi^2 \\ 0 & 0 & 1 & 0 & 0 & 1\,000\,000\,000\,000\,000\,\pi^4 \\ 0 & 0 & 0 & 1 & 0 & 1\,000\,000\,000\,000\,000\,\pi^6 \\ 0 & 0 & 0 & 0 & 1 & 1\,000\,000\,000\,000\,000\,V \end{pmatrix}$$

where V is a numeric approximation for our integral:

```
V = NIntegrate[Sqrt[x] Log[x]^4/(1 - x)^4, {x, 0, 1, Infinity}, WorkingPrecision → 35]
```

```
7.0087205930232887298578531032695675
```

Reduce the basis

```
Round[N[B, 30]];
LatticeReduce[%] // N
```

```
{{0., 12., -1., 0., -3., -8.35394 × 10^-14},
 {2432., -2008., 3191., -239., -9085., 6553.88},
 {-2689., -2621., -8150., 912., -7754., -6656.71},
 {16 891., 300., -772., 47., 1450., 3237.83},
 {-2049., -701., 20 817., -2029., -9722., -20 206.8}}
```

Choosing the shortest vector, yields

```
Clear[V];
Rationalize[%%[[1]]] .{1 , π^2 , π^4, π^6, V, 0}
```

$$12\,\pi^2 - \pi^4 - 3\,V$$

$$V = 4\,\pi^2 - \frac{\pi^4}{3}$$

■ **Example** (BBP formula for π)

Let us ask whether π satisfy a relation of the form

$$\sum_{k=0}^{\infty} \frac{1}{16^k}\left[\frac{a_1}{8\,k+1} + \frac{a_2}{8\,k+2} + \ldots + \frac{a_7}{8\,k+7}\right]$$

```
c = 10^15;
a[k_] = Sum[1/16^j * 1/(8 j + k), {j, 0, Infinity}];
```

```
B = {{1, 0, 0, 0, 0, 0, 0, 0, c a[1]},
    {0, 1, 0, 0, 0, 0, 0, 0, c a[2]},
    {0, 0, 1, 0, 0, 0, 0, 0, c a[3]},
    {0, 0, 0, 1, 0, 0, 0, 0, c a[4]},
    {0, 0, 0, 0, 1, 0, 0, 0, c a[5]},
    {0, 0, 0, 0, 0, 1, 0, 0, c a[6]},
    {0, 0, 0, 0, 0, 0, 1, 0, c a[7]},
    {0, 0, 0, 0, 0, 0, 0, 1, c Pi}};
B = Round[N[B, 30]];
```

Apply `LatticeReduce`

```
LatticeReduce[B] // N
```

```
{{-4., 0., 0., 2., 1., 1., 0., 1., 2.62812×10^-15},
 {0., -8., -4., -4., 0., 0., 1., 2., -2.10281×10^-16},
 {57., -15., -3., 17., 202., 12., -16., -30., -22.8166},
 {71., -71., 18., 89., 59., 75., -116., -23., 7.07971},
 {-17., 69., -51., -78., -83., 169., 8., 2., 1.89819},
 {32., 3., -58., 79., -133., 122., 88., -13., -53.1872},
 {-29., -20., 61., -26., -38., -29., -19., 13., -229.725},
 {36., -86., 142., 41., 24., 55., 130., -27., 61.3267}}
```

The first two vectors suggests two identities

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k}\left[\frac{4}{8\,k+1} - \frac{2}{8\,k+4} - \frac{1}{8\,k+5} - \frac{1}{8\,k+6}\right]$$

$$2\,\pi = \sum_{k=0}^{\infty} \frac{1}{16^k}\left[\frac{8}{8\,k+2} + \frac{4}{8\,k+3} + \frac{4}{8\,k+4} - \frac{1}{8\,k+7}\right]$$

The first formula is called the BBP-formula.

The significance of the BBP-formula is to compute far-out digits of π in the hexadecimal base. How can we do this?

A simpler problem:

given a rational number $\frac{a}{b}$, where $a < b$

compute a far-out digit (say 10^n th) of its decimal expansion

$$\frac{\text{Mod}[a\,\text{Mod}[10^n, b], b]}{b}$$

```
a = 1233219; b = 876543; n = 5; Mod[a Mod[10^n, b], b]/b // N
```

```
0.215377
```

```
N[a / b, 40]
```

```
1.406912153767698789449005924409869224898
```

Back to π. Suppose we want to compute digits starting at position $d+1$. You will need to multiple the above series by 16^d and take the fractional part. Let us demonstrate this by choosing the first additive term in the BBP formula

$$\text{frac}\left(16^d \sum_{k=0}^{\infty} \frac{4}{16^k (8k+1)}\right) = \text{frac}\left(16^d \sum_{k=0}^{d} \frac{4}{16^k (8k+1)}\right) + \text{frac}\left(16^d \sum_{k=d+1}^{\infty} \frac{4}{16^k (8k+1)}\right)$$

In the first sum

$$\text{frac}\left(16^d \sum_{k=0}^{d} \frac{4}{16^k (8k+1)}\right) = \sum_{k=0}^{d} \text{frac}\left(\frac{16^{d-k}}{8k+1}\right) = \sum_{k=0}^{d} \frac{16^{d-k} \,(\text{mod}\ 8k+1)}{8k+1} \,(\text{mod}\ 1)$$

1) we do exponentiation using the binary algorithm and reducing each intermediate product modulo $8k+1$;

2) divide each numerator by correspondent $8k+1$ using ordinary floating arithmetic;

3) sum terms discarding integer parts.

In the second sum

$$\text{frac}\left(\sum_{k=d+1}^{\infty} \frac{16^{d-k}}{8k+1}\right) = \sum_{k=d+1}^{\infty} \frac{16^{d-k}}{8k+1} \,(\text{mod}\ 1)$$

we will need only a few terms, since they rapidly become smaller. Adding these two sums together will yield a few digits of π starting at position $d+1$. See [8] for proofs and some computational details .

Concluding remarks

1) The lattice reduction algorithms do not find the shortest basis, but find a basis with the relatively short vectors.

2) LLL might run into numerical instability - you have to use "enough" digits.

3) The relation which you obtain is only a "possible" relation, it must be proved analytically!

4) The lattice reduction approach is very powerful and offers rich possibility for discovery!

References

[1] H. Ferguson and R. Forcade, Generalization of the Euclidean Algorithm for Real Numbers to All Dimensions Higher than Two, *Bull. Amer. Math. Soc.* , **1**(1979), 912-914.

[2] A. K. Lenstra, H. W. Lenstra Jr., L. Lovász, Factoring polynomials with rational coefficients. *Math. Ann.* **261** (1982), 515-534.

[3] H. Ferguson, D. Bailey, A Polynomial Time, Numerically Stable Integer Relation Algorithm, RNR Techn. Rept. RNR-91-032, Jul. 14, 1992.

[4] J. Håstad, B. Just, J. C. Lagarias, C. P. Schnorr, Polynomial time algorithms for finding integer relations among real numbers. *SIAM J. Comput.* **18** (1989), 859-881.

[5] J. Borwein, P. Lisonek, Applications of integer relation algorithms. *Discrete Mathematics*, **217** (2000), 65-82

[6] D.. Bailey, P.. Borwein and S. Plouffe, On the Rapid Computation of Various Polylogarithmic Constant, *Mathematics of Computation*, **66** (1997), 903-913

[7] J. Borwein, D, Bailey and R. Girgensohn, *Experimentation in Mathematics: Computational Paths to Discovery*, AK Peters Ltd, 2003.

[8] V. Adamchik, S. Wagon, π: A 2000-Year Search Changes Direction, *Mathematica in Education and Research*, no.1, **5**(1996) 11-19.