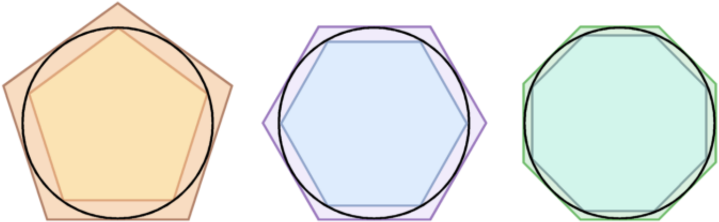
Taking precision to the limit

Fredrik Johansson

Unithé ou café, Inria Bordeaux-Sud-Ouest
November 2015

Archimedes (c. 287 BC – c. 212 BC) and π



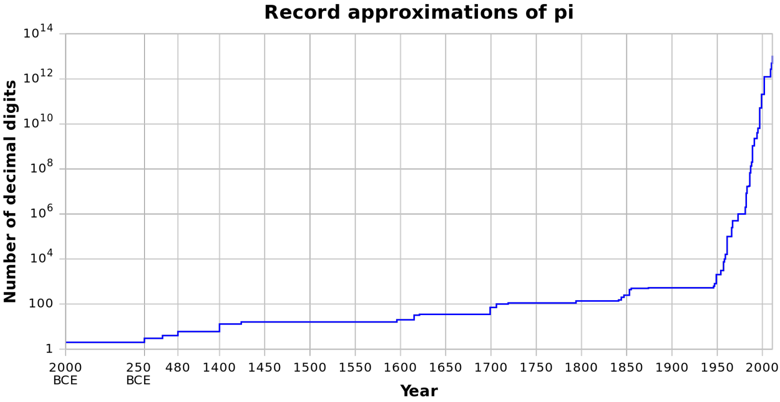
Using 96-sided polygons, Archimedes proved that

$$3 + \frac{10}{71} < \pi < 3 + \frac{1}{7}$$

or in decimal notation

$$3.140845\ldots < \pi < 3.142857\ldots$$

2000 years of progress. . .



http://en.wikipedia.org/wiki/Chronology_of_computation_of_pi

From hand calculation to computers

In 1873, William Shanks finished a calculation of π to **707 digits** which had taken him 15 years to complete.

From hand calculation to computers

In 1873, William Shanks finished a calculation of π to **707 digits** which had taken him 15 years to complete.

The record stood until 1946.

From hand calculation to computers

In 1873, William Shanks finished a calculation of π to **707 digits** which had taken him 15 years to complete.

The record stood until 1946.

In fact, only the first 527 digits were correct.

From hand calculation to computers

In 1873, William Shanks finished a calculation of π to **707 digits** which had taken him 15 years to complete.

The record stood until 1946.

In fact, only the first 527 digits were correct.

Since the age of digital computers, new records have followed the increase in computing capacity. Today, a mobile phone can compute **1 million digits** of π in a second.

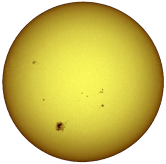
The current record is 13 300 000 000 000 digits, set by Alexander Yee and Shigeru Kondo in 2014.

Use for 16 digits

16-digit precision is standard in software for scientific calculations, and supported very efficiently in computer hardware.

Use for 16 digits

16-digit precision is standard in software for scientific calculations, and supported very efficiently in computer hardware.



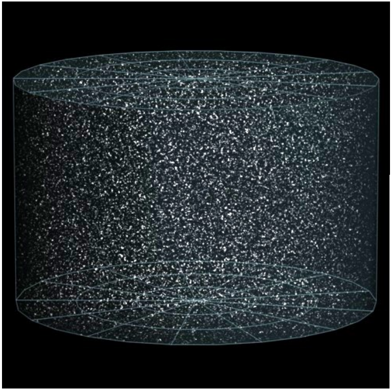
Not shown to scale.

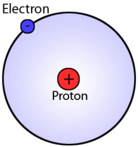
This is enough to resolve the distance between **Earth and the Sun** (150 000 000 km) to within the **width of a hair** (0.1 mm).

Orbital length $C = 2\pi r$

Use for 40 digits

40 digits allows resolving the **observable universe** (10^{27} m) to the level of detail of a single **hydrogen atom** (10^{-10} m).





Not shown to scale.

Isn't higher precision completely *useless*?

Isn't higher precision completely *useless*?

Example of a **useful** invention: *Cyclomer*, bicycle that can be used both on land and in water, Paris, 1932.

Why higher precision is sometimes needed

1. When doing a sequence of approximate calculations, small errors sometimes quickly add up to much larger errors

Why higher precision is sometimes needed

1. When doing a sequence of approximate calculations, small errors sometimes quickly add up to much larger errors
2. Extremely large or small numbers sometimes appear in the mathematical methods used to solve a problem

Why higher precision is sometimes needed

1. When doing a sequence of approximate calculations, small errors sometimes quickly add up to much larger errors
2. Extremely large or small numbers sometimes appear in the mathematical methods used to solve a problem

Some examples from recent research in mathematical physics and number theory (Bailey and Borwein, 2015):

- Long-term stability of the solar system: **30 digits**

Why higher precision is sometimes needed

1. When doing a sequence of approximate calculations, small errors sometimes quickly add up to much larger errors
2. Extremely large or small numbers sometimes appear in the mathematical methods used to solve a problem

Some examples from recent research in mathematical physics and number theory (Bailey and Borwein, 2015):

- Long-term stability of the solar system: **30 digits**
- Coulomb *n*-body atomic systems: **120 digits**

Why higher precision is sometimes needed

1. When doing a sequence of approximate calculations, small errors sometimes quickly add up to much larger errors
2. Extremely large or small numbers sometimes appear in the mathematical methods used to solve a problem

Some examples from recent research in mathematical physics and number theory (Bailey and Borwein, 2015):

- Long-term stability of the solar system: **30 digits**
- Coulomb *n*-body atomic systems: **120 digits**
- Zeros of the Riemann zeta function: **10 000 digits**

Why higher precision is sometimes needed

1. When doing a sequence of approximate calculations, small errors sometimes quickly add up to much larger errors
2. Extremely large or small numbers sometimes appear in the mathematical methods used to solve a problem

Some examples from recent research in mathematical physics and number theory (Bailey and Borwein, 2015):

- Long-term stability of the solar system: **30 digits**
- Coulomb *n*-body atomic systems: **120 digits**
- Zeros of the Riemann zeta function: **10 000 digits**
- Lattice sums arising from the Poisson equation: **50 000 digits**

The partition number problem

A *partition* of a number n is a representation of n as a sum of positive whole numbers.

The partition number problem

A *partition* of a number n is a representation of n as a sum of positive whole numbers.

For example, $3 + 1 + 1$ is a partition of $n = 5$.

The partition number problem

A *partition* of a number n is a representation of n as a sum of positive whole numbers.

For example, $3 + 1 + 1$ is a partition of $n = 5$.

Question: given the number n, how many different partitions of n are there?

Example

$=$

5

Example

$=$

$=$

5 $\qquad$ $4+1$

Example

$=$

5

$=$

$4 + 1$

$=$

$3 + 2$

Example

$=$

5

$=$

$4 + 1$

$=$

$3 + 2$

$=$

$3 + 1 + 1$

Example

$=$  $=$  $=$ 

5 $\qquad 4+1$ $\qquad 3+2$

$=$  $=$ 

$3+1+1$ $\qquad 2+2+1$

Example

$=$

5

$=$

$4 + 1$

$=$

$3 + 2$

$=$

$3 + 1 + 1$

$=$

$2 + 2 + 1$

$=$

$2 + 1 + 1 + 1$

Example

=

5

=

$4 + 1$

=

$3 + 2$

=

$3 + 1 + 1$

=

$2 + 2 + 1$

=

$2 + 1 + 1 + 1$

=

$1 + 1 + 1 + 1 + 1$

There are 7 different partitions of 5.

Larger numbers...

There are 7 partitions of 5

There are 42 partitions of 10

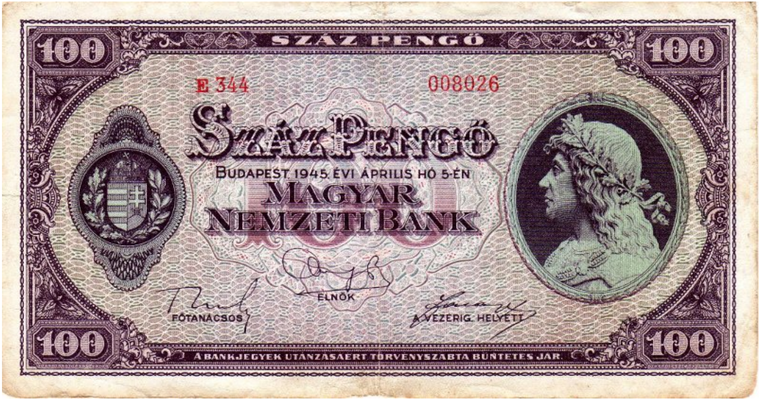
There are 627 partitions of 20

There are 204226 partitions of 50

There are 190569292 partitions of 100

Hungarian pengő (1 P, 1926)

10^2 pengő (April 1945)



100 pengő

10^3 pengő (July 1945)

1000 pengő

10^4 pengő (July 1945)

10 000 pengő

10^5 pengő (October 1945)

100 000 pengő

10^6 pengő (November 1945)

1 000 000 pengő

10^7 pengő (November 1945)

10 000 000 pengő

10^8 pengő (March 1946)

100 000 000 pengő

10^9 pengő (March 1946)

1 000 000 000 pengő

10^{10} pengő (April 1946)

10 000 milpengő = 10 000 000 000 pengő

10^{11} pengő (April 1946)

100 000 milpengő = 100 000 000 000 pengő

10^{12} pengő (May 1946)

$1\,000\,000$ milpengő $= 1\,000\,000\,000\,000$ pengő

10^{13} pengő (May 1946)

10 000 000 milpengő = 10 000 000 000 000 pengő

10^{14} pengő (June 1946)

100 000 000 milpengő = 100 000 000 000 000 pengő

10^{15} pengő (June 1946)

1 000 000 000 milpengő = 1 000 000 000 000 000 pengő

10^{16} pengő (June 1946)

10 000 b.-pengő = 10 000 000 000 000 000 pengő

10^{17} pengő (June 1946)

100 000 b.-pengő = 100 000 000 000 000 000 pengő

10^{18} pengő (June 1946)

1 million b.-pengő = 1 000 000 000 000 000 000 (1 quintillion) pengő

10^{19} pengő (June 1946)

10 million b.-pengő = 10 000 000 000 000 000 000 (10 quintillion) pengő

10^{20} pengő (June 1946)

100 million b.-pengő = 100 000 000 000 000 000 000 (100 quintillion) pengő

August 1946

On July 31, 10^{20} pengő was worth \$0.0000000002 USD.

On August 1, Hungary switched to a new currency, the forint.

Partitions of 10^{20}

In how many ways can you make change for a 10^{20}-pengő banknote using stacks of 1-pengő coins?

Partitions of 10^{20}

In how many ways can you make change for a 10^{20}-pengő banknote using stacks of 1-pengő coins?

$=$

10^{20}

Partitions of 10^{20}

In how many ways can you make change for a 10^{20}-pengő banknote using stacks of 1-pengő coins?

$=$

10^{20}

$=$

$99999999999999999999 + 1$

$\dots$

Partitions of 10^{20}

In how many ways can you make change for a 10^{20}-pengő banknote using stacks of 1-pengő coins?

$=$

10^{20}

$=$

$99999999999999999999 + 1$

$\cdots$

Practical difficulty: a stack of 10^{20} coins would reach from Earth to Gliese 581d, an extrasolar planet 20 light years way. If spread out over France, the coins would reach 100 meters high!

The answer

The exact number of partitions of 10^{20} is

$$\underbrace{1838176508344882\ldots 231756788091448}_{11\,140\,086\,260 \text{ digits}}$$

In scientific notation, this is approximately $1.8 \cdot 10^{11140086259}$

The answer

The exact number of partitions of 10^{20} is

$$\underbrace{1838176508344882\ldots231756788091448}_{11\,140\,086\,260 \text{ digits}}$$

In scientific notation, this is approximately $1.8 \cdot 10^{11140086259}$

I computed the exact number in 2014. It took **five days**, using two CPU cores and 130 GB of memory.

The answer

The exact number of partitions of 10^{20} is

$$\underbrace{18381765083448 82\ldots 231756788091448}_{11\,140\,086\,260 \text{ digits}}$$

In scientific notation, this is approximately $1.8 \cdot 10^{11140086259}$

I computed the exact number in 2014. It took **five days**, using two CPU cores and 130 GB of memory.

The method is to deduce the exact value from an extremely precise **numerical approximation**. No known methods that only use exact steps would have been feasible!

From approximate to exact

Example: how many partitions are there of $n = 100$?

$$190568944.78 + 348.871 - 2.60 + 0.685 + 0.318 \approx 190569292.06$$

From approximate to exact

Example: how many partitions are there of $n = 100$?

$$190568944.78 + 348.871 - 2.60 + 0.685 + 0.318 \approx 190569292.06$$

For $n = 10^{20}$, we need to add $1\,710\,193\,158$ terms, with a precision of more than 11 billion digits!

From approximate to exact

Example: how many partitions are there of $n = 100$?

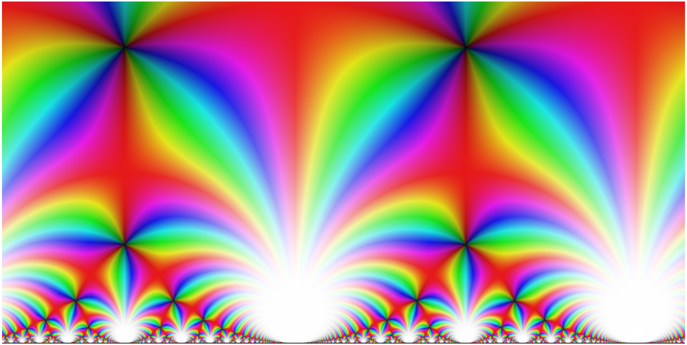
$$190568944.78 + 348.871 - 2.60 + 0.685 + 0.318 \approx 190569292.06$$

For $n = 10^{20}$, we need to add 1 710 193 158 terms, with a precision of more than 11 billion digits!

Question: Where do the approximate numbers come from?

Question: How can we be sure that the result is correct?

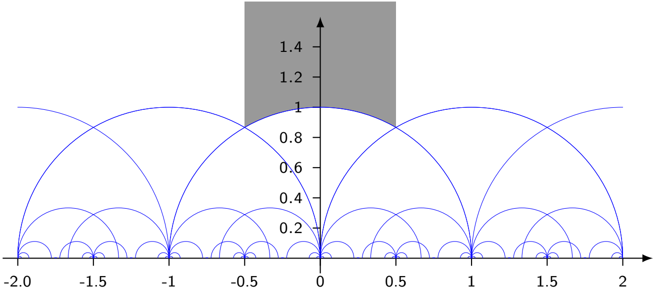
Modular forms

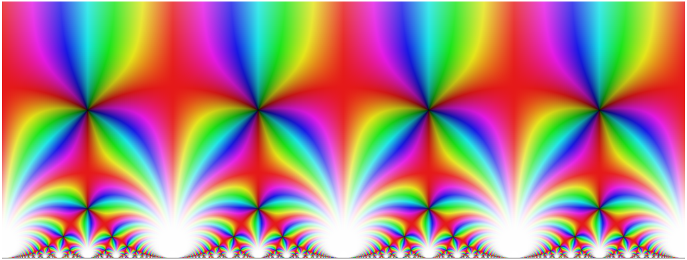


Modular forms are among the most important objects studied in modern number theory. The mathematician Martin Eichler (1912–1992) is supposed to have said:

> *There are five elementary arithmetical operations: addition, subtraction, multiplication, division, and... modular forms.*

Geometry of modular forms





Connection between modular forms and partitions

Leonhard Euler (1707–1783) observed that

$$\frac{1}{(1-x)(1-x^2)(1-x^3)(1-x^4)(1-x^5)\cdots}$$

$$= 1 + x + 2x^2 + 3x^3 + 5x^4 + 7x^5 + 11x^6 + 15x^7 + \ldots$$

The number in front of x^n is the number of partitions of n.

In modern terminology, $f(x) = (1-x)(1-x^2)(1-x^3)\cdots$ is the modular form which generates the partition numbers.

The circle method

G. H. Hardy and Srinivasa Ramanujan (1916) developed an ingenious way to use the **geometry** of the modular form to approximate the **numbers it generates**.

The circle method

G. H. Hardy and Srinivasa Ramanujan (1916) developed an ingenious way to use the **geometry** of the modular form to approximate the **numbers it generates**.

Hans Rademacher (1943) proved that the *exact numbers* can be determined when the approximation is made precise enough.

The circle method

G. H. Hardy and Srinivasa Ramanujan (1916) developed an ingenious way to use the **geometry** of the modular form to approximate the **numbers it generates**.

Hans Rademacher (1943) proved that the *exact numbers* can be determined when the approximation is made precise enough.

Later generalized to many other types of counting problems.

The circle method

G. H. Hardy and Srinivasa Ramanujan (1916) developed an ingenious way to use the **geometry** of the modular form to approximate the **numbers it generates**.

Hans Rademacher (1943) proved that the *exact numbers* can be determined when the approximation is made precise enough.

Later generalized to many other types of counting problems.

In 2012, I published a detailed algorithmic analysis (and fast implementation) of this method for counting partitions.

Why care about counting partitions?

By *looking at tables of computed values*, Ramanujan discovered patterns. For example:

n	Number of partitions	n	Number of partitions
1	1	11	56
2	2	12	77
3	3	13	101
4	**5**	**14**	**135**
5	7	15	176
6	11	16	231
7	15	17	297
8	22	18	385
9	**30**	**19**	**490**
10	42	20	627

This observation opened up a major field of research within number theory which is still active.

More uses for modular forms

- Counting things (such as partitions)
- Mathematical proofs (example: Fermat's Last Theorem, $a^n + b^n = c^n$ has no integer solutions for $n > 2$)
- Algorithms for prime numbers
- Cryptography

A real-world application: elliptic curve cryptography

Security properties of an elliptic curve are encoded (in a complicated way) by a list of numbers called a *class polynomial*.

Modular form

$$\downarrow$$

Approximate class polynomial

$$(1, 39491306.99\ldots, -58682638134.02\ldots, 1566028350940383.01\ldots)$$

$$\downarrow$$

Exact class polynomial

$$(1, 39491307, -58682638134, 1566028350940383)$$

$$\downarrow$$

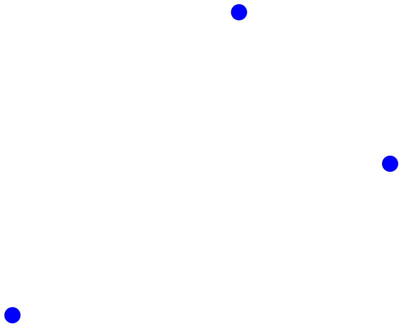
Proof of security

A class polynomial can have thousands of numbers, each of which has thousands of digits!

Numerical computing

Exact points

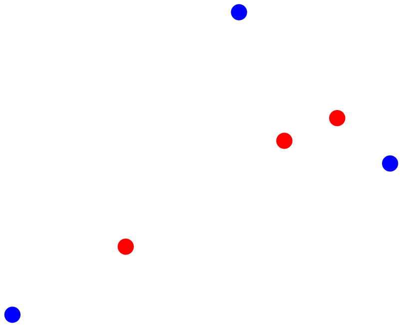
Computed approximations



Numerical computing

Exact points

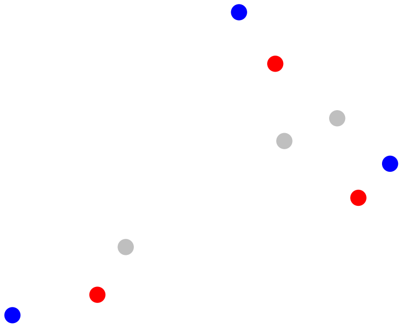
Computed approximations



Numerical computing

Exact points

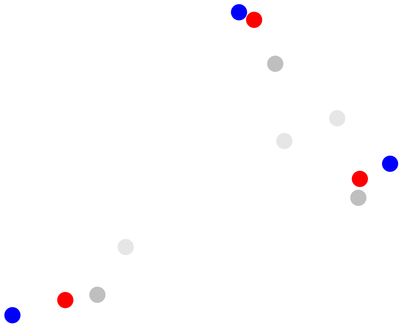
Computed approximations



Numerical computing

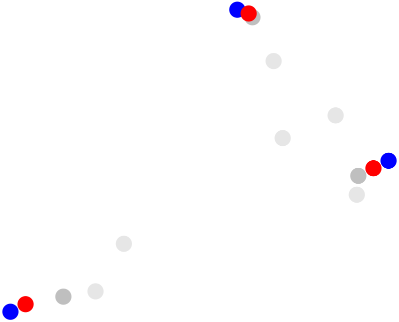
Exact points

Computed approximations



Numerical computing

Exact points
Computed approximations

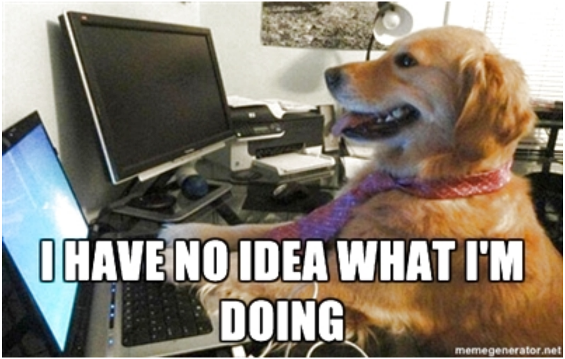


How do we know when we are “close enough”?

Method 1: guess and hope for the best

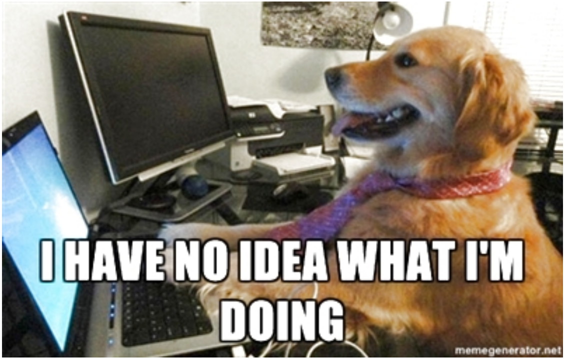
How do we know when we are “close enough”?

Method 1: guess and hope for the best



How do we know when we are "close enough"?

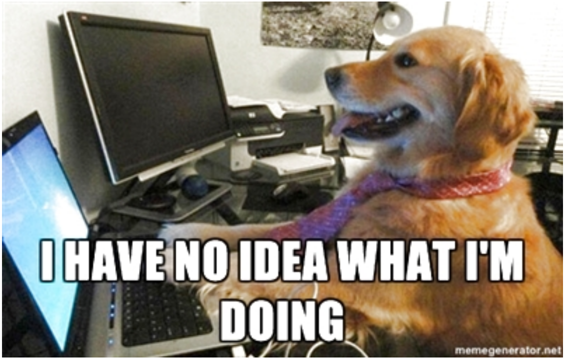
Method 1: guess and hope for the best



Method 2: error analysis

How do we know when we are “close enough”?

Method 1: guess and hope for the best



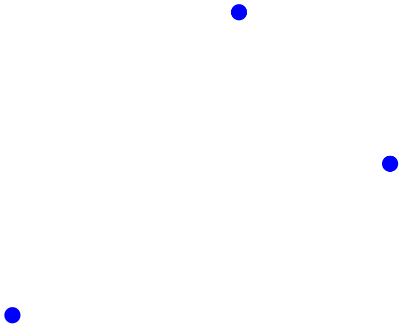
Method 2: error analysis

Method 3: interval arithmetic

Interval arithmetic

Exact points

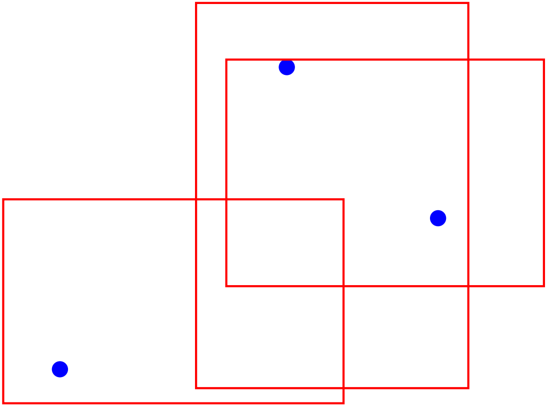
Computed enclosures



Interval arithmetic

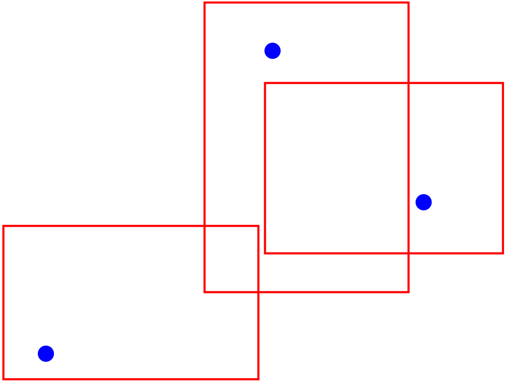
Exact points

Computed enclosures



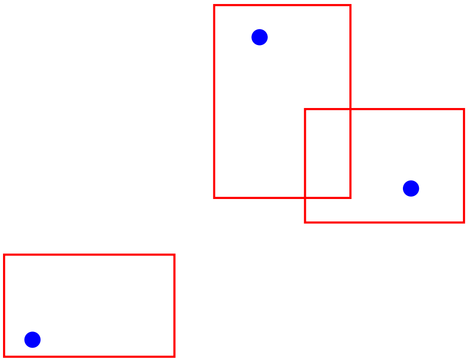
Interval arithmetic

Exact points
Computed enclosures



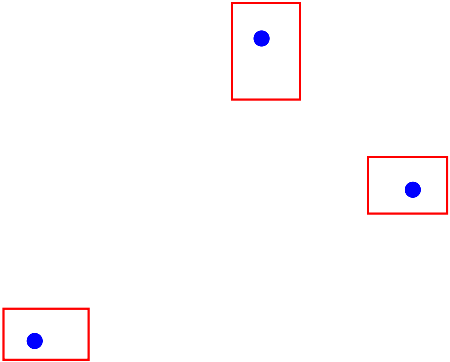
Interval arithmetic

Exact points
Computed enclosures



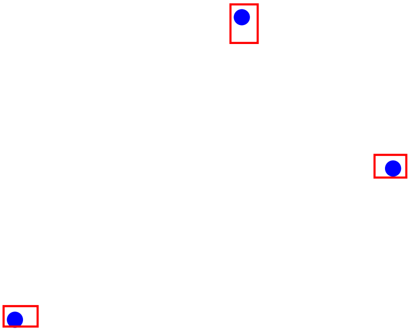
Interval arithmetic

Exact points
Computed enclosures

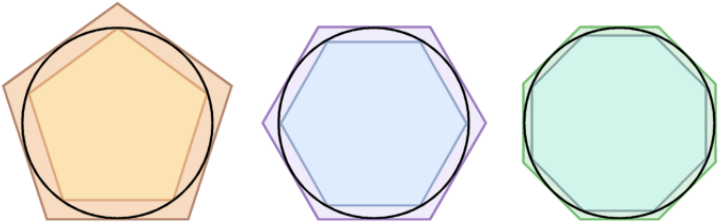


Interval arithmetic

Exact points
Computed enclosures



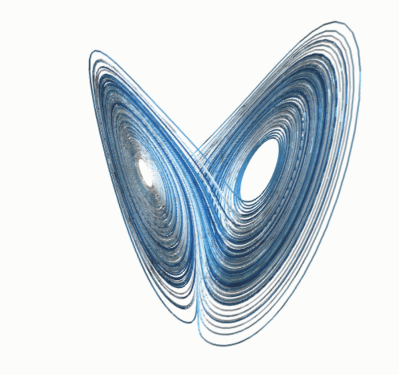
Archimedes revisited



Modern methods to compute π are much more efficient. In fact, the fastest methods are based on modular forms!

The Lorenz attractor

The *Lorenz attractor* is a famous example of a *chaotic system*.



For many years, no one knew whether the Lorenz attractor *exists*.
In 1998, Stephen Smale named this one of the most challenging mathematical problems for the next century.

In 2002, Warwick Tucker proved the existence. His proof made extensive use of interval arithmetic calculations.

The Kepler conjecture

In 1611, Johannes Kepler conjectured that the highest density that can be achieved when packing spheres is $\pi/(3\sqrt{2}) \approx 0.74$.

The Kepler conjecture was finally proved in 1998 by Thomas Hales, using extensive interval arithmetic calculations.

My research

Developing **efficient, reliable and general-purpose** algorithms and software for high-precision arithmetic.

Challenge: manipulating high-precision numbers efficiently. Large numbers must be stored as lists of smaller numbers, for example $314159265358979 = [314, 159, 265, 358, 979]$.

Challenge: methods for computing complex mathematical functions (such as modular forms).

Software

My software for high-precision interval arithmetic (Arb):

- `https://github.com/fredrik-johansson/arb/`
- Includes code for counting partitions, evaluating modular forms, and many other things.
- Currently being added to the SageMath computer algebra system `http://sagemath.org/`

Other software (+many others):

- MPFR: high-precision floating-point arithmetic `http://www.mpfr.org/`
- MPFI: high-precision interval arithmetic `https://perso.ens-lyon.fr/nathalie.revol/software.html`
- Pari/GP: `http://pari.math.u-bordeaux.fr/`

Take-home message

1. Some applications require computing with a precision of 100, 1000, 10000 or more digits.

2. For some mathematical problems, the best way to get an exact solution is to compute an approximate solution.

3. Interval arithmetic is useful – a mathematically rigorous way to do numerical computations.

The end

There are 10^{11} *stars in the galaxy. That used to be a huge number. But it's only a hundred billion. It's less than the national deficit! We used to call them astronomical numbers. Now we should call them economical numbers.*

– Richard Feynman

Image credits

- https://en.wikipedia.org/wiki/Chronology_of_computation_of_%CF%80
- https://commons.wikimedia.org/wiki/File:Sun920607.jpg
- https://commons.wikimedia.org/wiki/File:The_Earth_seen_from_Apollo_17.jpg
- https://commons.wikimedia.org/wiki/File:8_Observable_Universe_%28ELitU%29-blank.png
- https://commons.wikimedia.org/wiki/File:Bohr_model.jpg
- https://commons.wikimedia.org/wiki/File:Amfibiefiets_Amphibious_bicycle.jpg
- https://en.wikipedia.org/wiki/Hungarian_peng%C5%91
- https://commons.wikimedia.org/wiki/File:Lorenz.png
- https://commons.wikimedia.org/wiki/File:HCP_Oranges.jpg
- https://tex.stackexchange.com/questions/113677/tikz-and-fundamental-domains