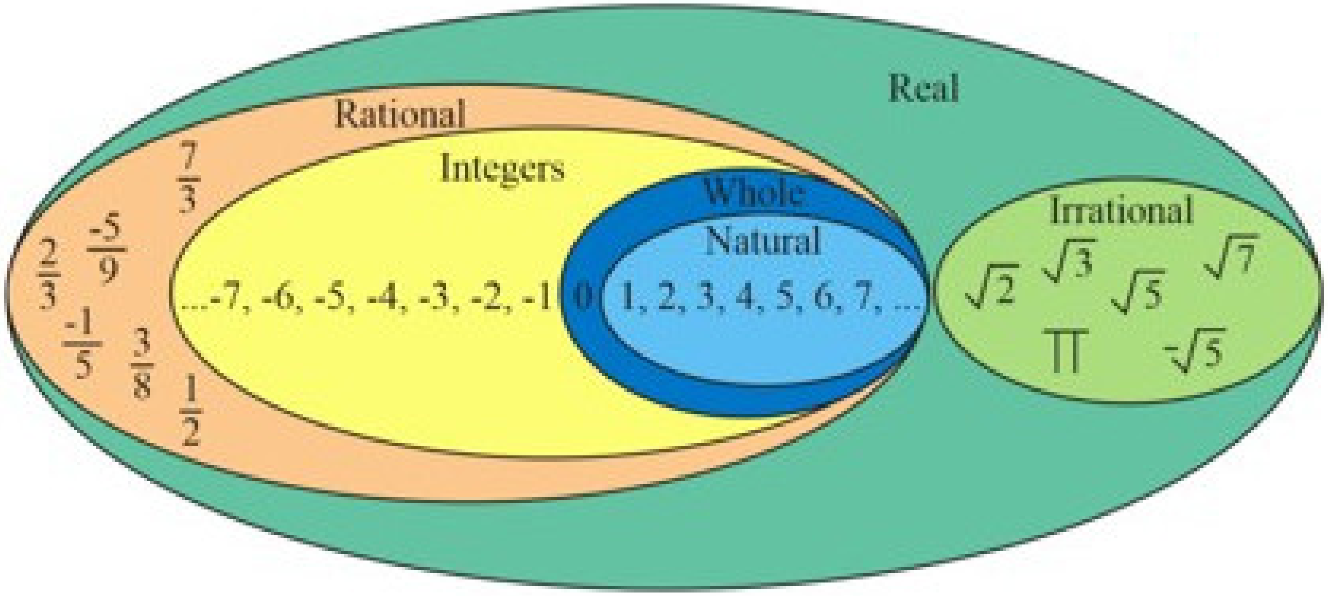
On Machine Integers and Arithmetic



Pavel Loskot
pavelloskot@intl.zju.edu.cn

ZJU-UIUC INSTITUTE
Zhejiang University-University of Illinois at Urbana-Champaign Institute
浙江大学伊利诺伊大学厄巴纳香槟校区联合学院

The Eighth International Conference on Advances in Signal, Image and Video Processing
SIGNAL 2023
March 13, 2023 to March 17, 2023 - Barcelona, Spain
IARIA

About Me

Pavel Loskot joined the ZJU-UIUC Institute as Associate Professor in January 2021. He received his PhD degree in Wireless Communications from the University of Alberta in Canada, and the MSc and BSc degrees in Radioelectronics and Biomedical Electronics, respectively, from the Czech Technical University of Prague. He is the Senior Member of the IEEE, Fellow of the HEA in the UK, and the Recognized Research Supervisor of the UKCGE.

In the past 25 years, he was involved in numerous industrial and academic collaborative projects in the Czech Republic, Finland, Canada, the UK, Turkey, and China. These projects concerned mainly wireless and optical telecommunication networks, but also genetic regulatory circuits, air transport services, and renewable energy systems. This experience allowed him to truly understand the interdisciplinary workings, and crossing the disciplines boundaries.

His current research focuses on statistical signal processing and importing methods from Telecommunication Engineering and Computer Science to model and analyze systems more efficiently and with greater information power.

Objectives

- a few fundamental claims about computing machines with finite memory
- a new look at the Fermat last theorem (FLT) as a side effect
- a work in progress, the proofs of claims not provided, but identified a number of research problems to explore

Outline

- Number Representations
- Integer Arithmetic
- Modular Arithmetic and Dual Modulo Operator
- Fermat Last Theorem and Fermat Metric

Number Representations

Number systems

- abstract mathematical objects (sets, groups, rings, fields, ...)
  → algebra, algebraic laws
- or carry semantic meaning of quantity
  → arithmetic operations, computing

Practical computing machines

- must represent numbers effectively within a finite memory
  → CPU registers, RAM
- numbers (i.e. variables) are pre-allocated finite space in memory
  → fixed and floating point representations with single and double precision
  → IEEE 754 standard
- unlimited precision is possible, but inefficient
  → GNU MP library (used by Mathematica and Maple)

Claim 1

Any practical computing only involves a finite set of computable numbers, $\mathcal{N} = \{N_1 < N_2 < \ldots\}$*. The differences,* $\min_{i\neq j} |N_i - N_j| = \epsilon_0$*, define the precision.*

Number Representations (cont.)

String representation

$$N = \sum_{i=i_{\min}}^{i_{\max}} D_i \times B^i \quad \leftrightarrow \quad D_{i_{\max}} D_{i_{\max}-1} \cdots D_1 D_0 \,.\, D_{-1} \cdots D_{i_{\min}}$$

- practical, human-readable representation
  → automata (e.g. Turing machine) and language theory
- decimal point split the string into integral and fractional parts
  → it is customary to insert decimal point between digits D_0 and D_{-1}
  → different bases, B, are mathematically fully equivalent
  → but they are not equivalent in operations with strings

Internal representation

- in-memory, representation as $B = 2^{8\times\#\text{bytes}-1}$ for efficiency reasons

Claim 2

Decimal point has purely syntactical meaning to align numbers in arithmetic operations. Consequently, all numbers on computing machines can be considered to be integers.

Integer Arithmetic

Finite set of integers

- overflow and underflow problems
- approximation of real-valued arithmetic and models
  → truncation, rounding
- aligning numbers in arithmetic operations
  → unified placement of decimal point
  → padding with zeros (from left and right)

Comparing numbers

- $0.99999\cdots9$ vs. $1.00000\cdots0$ problem
- tolerating the difference in scale induces periodicity
  → left-end sub-string
- tolerating the difference in precision induces quantization
  → right-end sub-string
- adjusting scale and/or precision can be done by modulo operator
  → assuming finite-length strings with zero-paddings

Modular Arithmetic

Canonical modulo operator

- for $b \in \mathbb{Z}$ or $b \in \mathcal{R}$

$$(a \bmod b) = (|a| \bmod b) \in \{0, 1, \ldots, b-1\}$$

- congruence vs. equality

$$a_1 \equiv a_2 \pmod{b}, \quad a_1 = a_2$$

→ reflexivity, symmetry, and transitivity
→ equality implies equivalence

Dual-modulo operator

- for $m_1 = B^{L-L_1}$ and $m_2 = B^{L_2}$, define

$$\begin{aligned} N_i \operatorname{Mod}(m_1, m_2) &= (N_i \bmod m_1) - (N_i \bmod m_2) \\ &= \underbrace{0 \cdots 0}_{L_1} D_{L-L_1-1} \cdots D_{L_2+1} D_{L_2} \underbrace{0 \cdots 0}_{L_2} \end{aligned}$$

- similar properties as canonical modulo operator

Modular Arithmetic (cont.)

Properties of dual-modulo operator

$$a\ \mathrm{Mod}(0,m_2) = a-(a \bmod m_2)$$

$$a\ \mathrm{Mod}(m_1,1) = a \bmod m_1$$

$$a\ \mathrm{Mod}(m_1,m_1) = 0$$

$$a+b \equiv a\ \mathrm{Mod}(m_1,m_2)+b\ \mathrm{Mod}(m_1,m_2)\ (\mathrm{Mod}(m_1,m_2))$$

$$a-b \equiv a\ \mathrm{Mod}(m_1,m_2)-b\ \mathrm{Mod}(m_1,m_2)\ (\mathrm{Mod}(m_1,m_2))$$

$$a\cdot b \equiv a\ \mathrm{Mod}(m_1,m_2)\cdot b\ \mathrm{Mod}(m_1,m_2)\ (\mathrm{Mod}(m_1,m_2))$$

$$a/b \not\equiv a\ \mathrm{Mod}(m_1,m_2)/b\ \mathrm{Mod}(m_1,m_2)\ (\mathrm{Mod}(m_1,m_2))$$

Chinese reminder-theorem

- if m_{11} and m_{12} are co-prime and, for some integers N_i and m_2

$$N_i \equiv a_1\ (\mathrm{Mod}(m_{11},m_2)) \text{ and } N_i \equiv a_2\ (\mathrm{Mod}(m_{12},m_2))$$

- then there is a unique integer a such that

$$N_i \equiv a\ (\mathrm{Mod}(m_{11}m_{12},m_2))$$

Case Study: FLT

Original formulation

- there are no integers $a,b,c,\in\mathbb{N}$, such that $a^n+b^n=c^n$, if $n>2$
- long and mathematically very evolved proof published recently

FLT re-formulation #1

- for every $n\in\mathbb{N}$, there exist infinitely many $(a,b,c,m_1,m_2)\in\mathbb{N}^5$, such that

$$a^n+b^n\equiv c^n\ (\mathrm{Mod}(m_1,m_2))$$

- example assuming numbers with $l=l_1+l_2=9$ digits and integers ≤ 100

	$B=8$				$B=10$			
	$n=3$		$n=4$		$n=3$		$n=4$	
(l_1,l_2)	$(3,6)$	$(4,5)$	$(3,6)$	$(4,5)$	$(3,6)$	$(4,5)$	$(3,6)$	$(4,5)$
n_1	69627	22278	5505	2318	1284	44532	10666	3622
n_r	212	644	730	2076	198	207	230	596

Case Study: FLT (cont.)

Define integers $\tilde{\mathbb{N}}_x$

- given a constant $x \in \mathcal{R}$, let

$$\tilde{\mathbb{N}}_x = \{x, x+1, x+2, \ldots\}$$

- satisfy Peano axioms, except $0 \to x$
  $\to \mathbb{N}_0$ are natural numbers
- however, $(a+b) \notin \tilde{\mathbb{N}}_x$ when $a, b \in \tilde{\mathbb{N}}_x$
  $\to$ still a good starting point for developing this further

FLT re-formulation #2

- for every $n \in \mathbb{N}$, there is always a solution $(a,b,c) \in \tilde{\mathbb{N}}_x^3$, such that

$$a^n + b^n = c^n$$

- thus, $(a-x)$, $(b-x)$, and $(c-x)$ are natural integers

Case Study: FLT (cont.)

FLT re-formulation #3

- for every $n \in \mathbb{N}$, there exists integer $m \geq n$, and a set of natural integers $\{a_1, a_2, \ldots, a_m\} \cup \{b\}$, such that

$$a_1^n + a_2^n + \cdots + a_m^n = b^n$$

- examples:

$$3^2 + 4^2 = 5^2 \ (m = n = 2)$$

$$3^3 + 4^3 + 5^3 = 6^3 \ (m = n = 3)$$

$$2^4 + 2^4 + 3^4 + 4^4 + 4^4 = 5^4 \ (m = n + 1 = 5)$$

$$19^5 + 43^5 + 46^5 + 47^5 + 67^5 = 72^5 \ (m = n = 5)$$

Fermat Metric

Observations

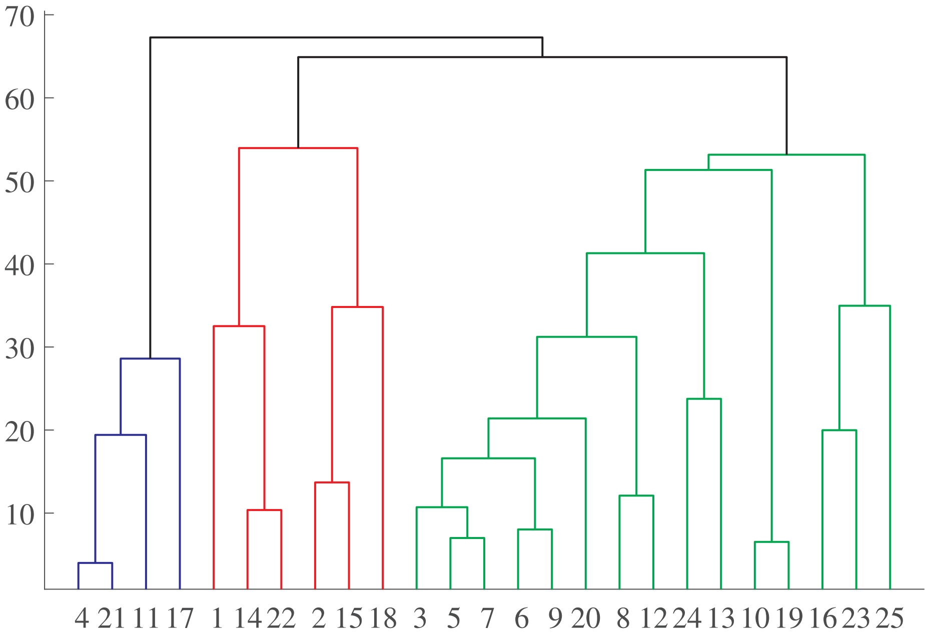
- sequence $a^n + b^n$ becomes sparse very rapidly with increasing n
- the best approximation of $(a^n + b^n)$ is by c^n where $c = \lfloor (a^n + b^n)^{1/n} \rceil$

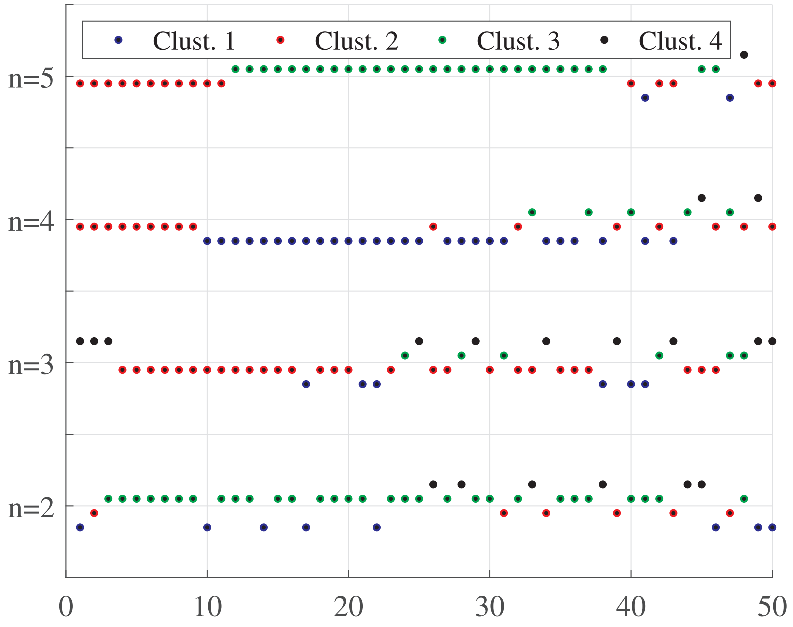
Definition

$$\mathcal{F}_n(a,b) = a^n + b^n - \lfloor (a^n + b^n)^{1/n} \rceil^n$$

- $\mathcal{F}_1(a,b) = 0$
- Fermat distance

$$D_n(a,b) = |\mathcal{F}_n(a,b)|$$





Conclusion

Claim 3

Any computing system utilizing finite number representations can be represented by a system of Diophantine equations.

Key points

- large gap between real-valued models and actual computer implementations
- computing models can/should exploit congruent equivalences
- FLT can be modified to allow the solutions to exist
- integers can be defined as $\tilde{\mathbb{N}}_x$
- Fermat distance allows clustering natural integers as well as real numbers

Future work

- arithmetic involving integers, $r_1 n + r_2$, where $n \in \mathbb{N}$ and $r_1, r_2 \in \mathcal{R}$
- obtaining partial proofs for some number theory problems
  → incomplete proofs which nevertheless have clear information value
- implications of integer arithmetic on constructing computing machines
  → including Turing machine
- applications of Fermat metric/distance, integers $\tilde{\mathbb{N}}_x$, etc.

Thank you!

pavelloskot@intl.zju.edu.cn