Some Unsolved and Unsolvable Problems in Mathematics

Uli Fahrenberg

EPITA Rennes, France

20 June 2021

ÉCOLE D'INGÉNIEURS EN INFORMATIQUE

The Collatz Problem

The Collatz Conjecture

Definition

The Collatz function f is defined on natural numbers, by

$$f(x) = \begin{cases} x/2 & \text{if } x \text{ is even} \\ 3x+1 & \text{if } x \text{ is odd} \end{cases}$$

Define $f_2(x) = f(f(x))$, $f_3(x) = f(f(f(x)))$; in general,

$$f_n(x) = \underbrace{f(f(f(\cdots(}_{n \text{ times}} x)\cdots)))$$

Collatz Conjecture

For every natural number x there is an index n for which $f_n(x) = 1$.

Collatz in Python

```
def collatz(x):
    if x % 2 == 0:
        return x // 2
    else:
        return 3 * x + 1

def collatz_sequence(x):
    while x != 1:
        print(x," ",end="")
        x = collatz(x)
    print(x)

collatz_sequence(27)
```

Collatz 27

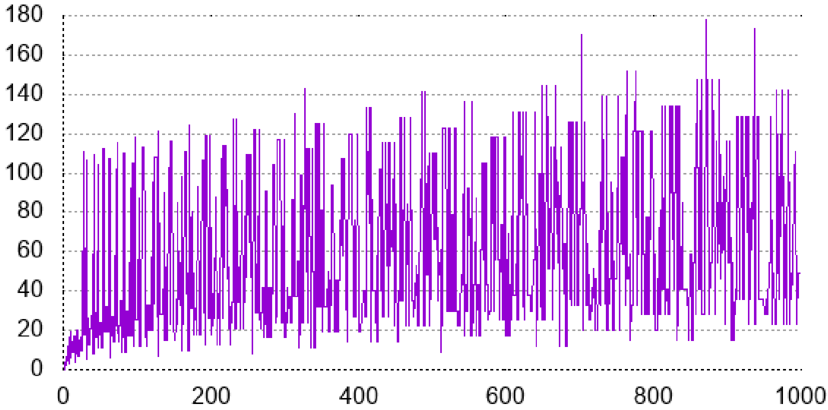
uli@sibelius: ~/Talks/2022/StAubin

```
uli@sibelius:~/Talks/2022/StAubin$ ./collatz1.py
27  82  41  124  62  31  94  47  142  71  214  107  322  161
  484  242  121  364  182  91  274  137  412  206  103  310
 155  466  233  700  350  175  526  263  790  395  1186  593
  1780  890  445  1336  668  334  167  502  251  754  377  1
132  566  283  850  425  1276  638  319  958  479  1438  719
  2158  1079  3238  1619  4858  2429  7288  3644  1822  911
 2734  1367  4102  2051  6154  3077  9232  4616  2308  1154
 577  1732  866  433  1300  650  325  976  488  244  122  61
  184  92  46  23  70  35  106  53  160  80  40  20  10  5
16  8  4  2  1
uli@sibelius:~/Talks/2022/StAubin$
```

Collatz Numbers

$$f(x) = \begin{cases} x/2 & \text{if } x \text{ is even} \\ 3x+1 & \text{if } x \text{ is odd} \end{cases}$$

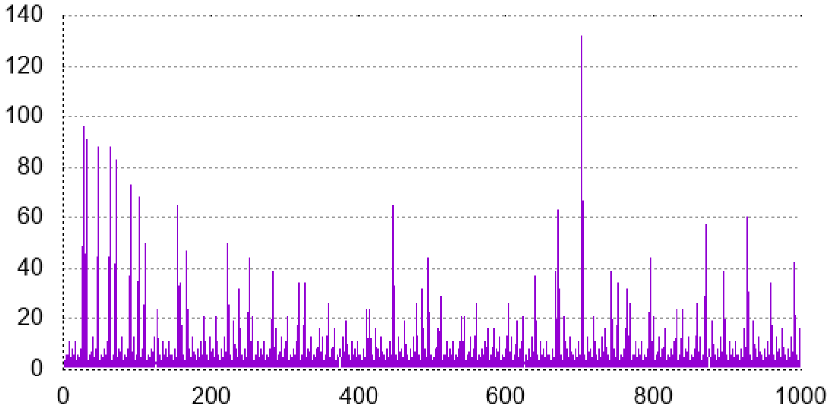
- the Collatz number of x: smallest index n for which $f_n(x) = 1$



Flying Times

$$f(x) = \begin{cases} x/2 & \text{if } x \text{ is even} \\ 3x+1 & \text{if } x \text{ is odd} \end{cases}$$

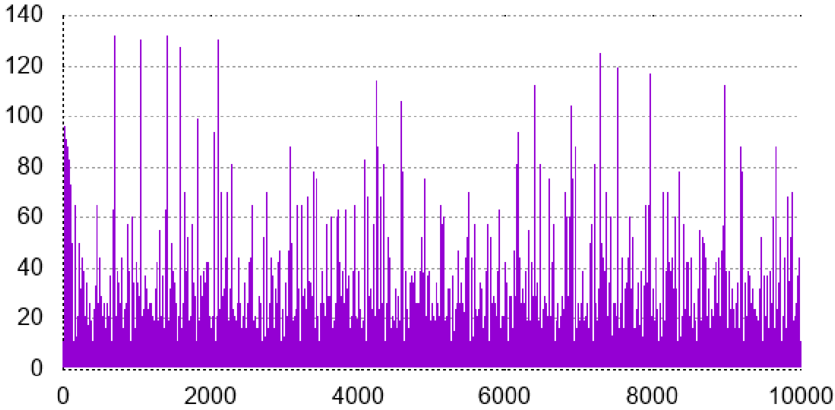
- the flying time of x: smallest index n for which $f_n(x) \leq x$



Flying Times

$$f(x) = \begin{cases} x/2 & \text{if } x \text{ is even} \\ 3x + 1 & \text{if } x \text{ is odd} \end{cases}$$

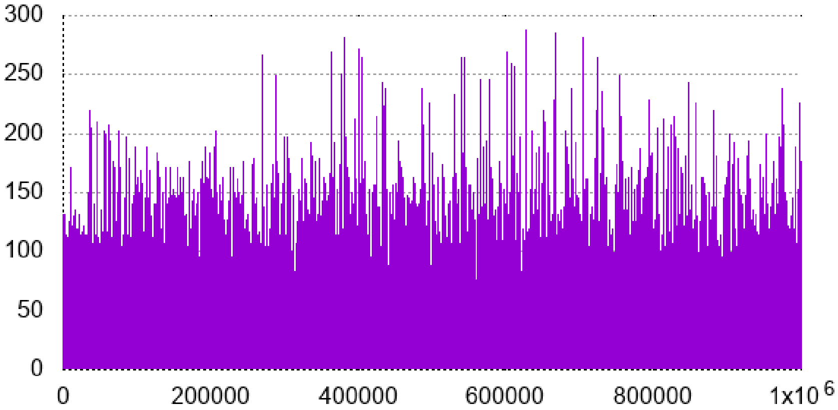
- the flying time of x: smallest index n for which $f_n(x) \leq x$



Flying Times

$$f(x) = \begin{cases} x/2 & \text{if } x \text{ is even} \\ 3x+1 & \text{if } x \text{ is odd} \end{cases}$$

- the flying time of x: smallest index n for which $f_n(x) \le x$



The Collatz Conjecture: Summing Up

Definition (recall)

The Collatz function f is defined on natural numbers, by

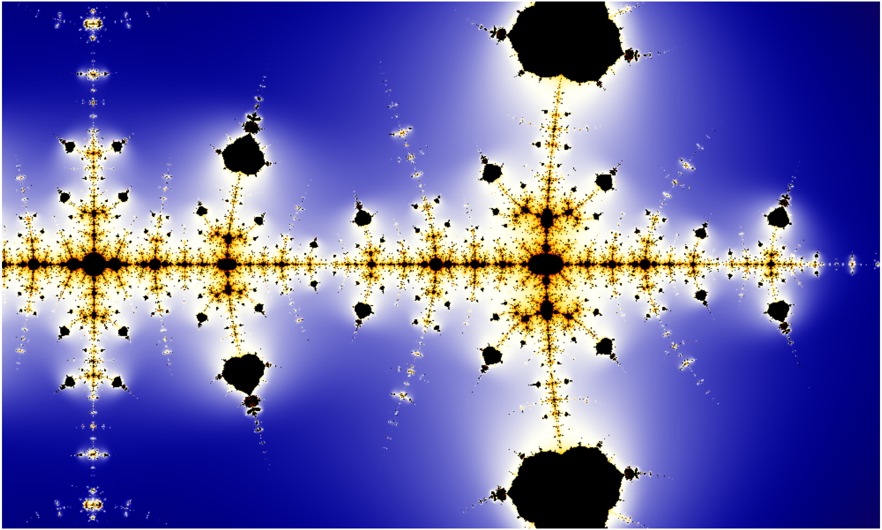
$$f(x) = \begin{cases} x/2 & \text{if } x \text{ is even} \\ 3x+1 & \text{if } x \text{ is odd} \end{cases}$$

Collatz Conjecture (recall)

For every natural number x there is an index n for which $f_n(x) = 1$.

- open since 1937
- "Mathematics may not be ready for such problems."
- "An extraordinarily difficult problem, completely out of reach of present day mathematics."

The Collatz Fractal



Part II

Unsolvable Problems

Programmers Make Mistakes

```
def collatz(x):
    if x % 2 == 0:
        return x // 2
    else:
        return 4 * x + 1

def collatz_sequence(x):
    while x != 1:
        print(x," ",end="")
        x = collatz(x)
    print(x)

collatz_sequence(27)
```

Programs Have Bugs

Fact of life

All programs contain mistakes (bugs).

How to find bugs?

- Can we write programs which can find bugs in other programs?
  - Yes! Plenty exist.
- Can we write programs which can find all bugs in other programs?
  - No!

The Halting Problem

Theorem

There exists no program which, when given another program Q as input, can decide whether Q ever finishes.

- So it is undecidable whether a program ever stops.
- That is, the *Halting problem* is unsolvable.
- (And so are plenty of other problems in mathematics!)

The Halting Problem: Proof

Theorem

There exists no program "`halts`" which, when given another program Q as input, can decide whether Q ever finishes.

Proof: Assume `halts` exists.

```
def liar():
    if halts(liar) == true:
        loop_forever()
```

Conclusion

- The Collatz problem: simple to state, but unsolved
- The Halting problem: simple to state, but unsolvable
- Uli Fahrenberg, EPITA Rennes: any questions?