ALGORITHMS

Richard Harvey

GRESHAM
COLLEGE

ALGORITHMS

Richard Harvey

IT Livery Company Professor of Information Technology, Gresham College

GRESHAM

COLLEGE

www.prof-richard.org

Arabic numerals

Muhammed al Khwarizmi
AD 780

Algebra

Algorithms

algorithm, n.

1. The Arabic system of numbering, characterized by a zero (cf ALGORISM n. 1; now *rare*). Formerly also: calculus (*obsolete*)

2. *Mathematics and Computing*, procedure or set of rules used in calculation and problem-solving; (in later use *spec.*) a precisely defined set of mathematical or logical operations for the performance of a particular task.

3. Medicine. A step-by-step protocol used to reach a clinical diagnosis or decision.

"algorithm, n." *OED Online*, Oxford University Press, September 2020, www.oed.com/view/Entry/4959. Accessed 23 September 2020

An *effective method*

A finite number of exact, finite instructions

When applied to problem from its class

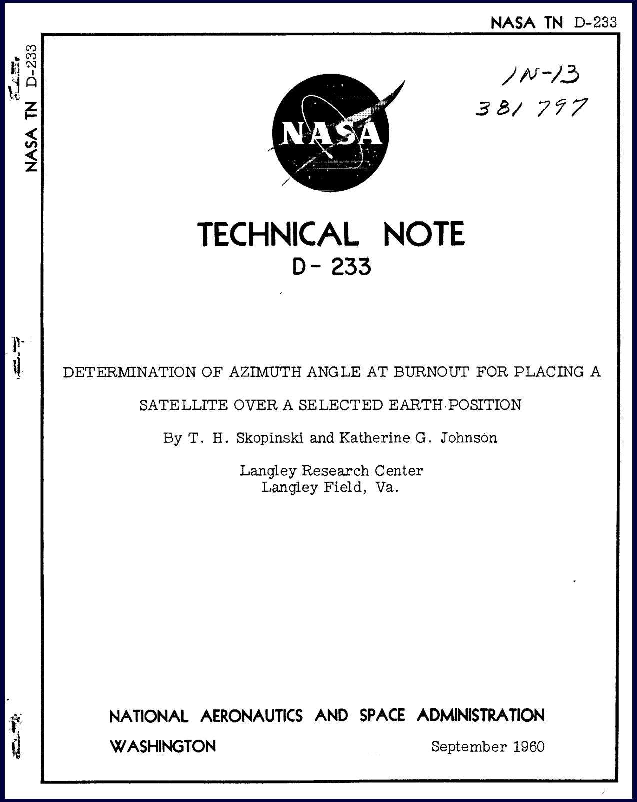
- Always stops after a finite number of steps
- Produces the correct answer
- In principle can be done by any human without any aids except writing materials
- Need only follow the instructions rigorously to succeed (no ingenuity required)

Algorithms are:

- Unambiguous (not the same as deterministic)
- Can be expressed in a finite amount of time and space

Algorithms are not programs

- Algorithms are converted into instructions that can be followed by computers.



NASA TN D-233

NASA TN D-233

IN-13
381 797

NASA

TECHNICAL NOTE
D-233

DETERMINATION OF AZIMUTH ANGLE AT BURNOUT FOR PLACING A SATELLITE OVER A SELECTED EARTH POSITION

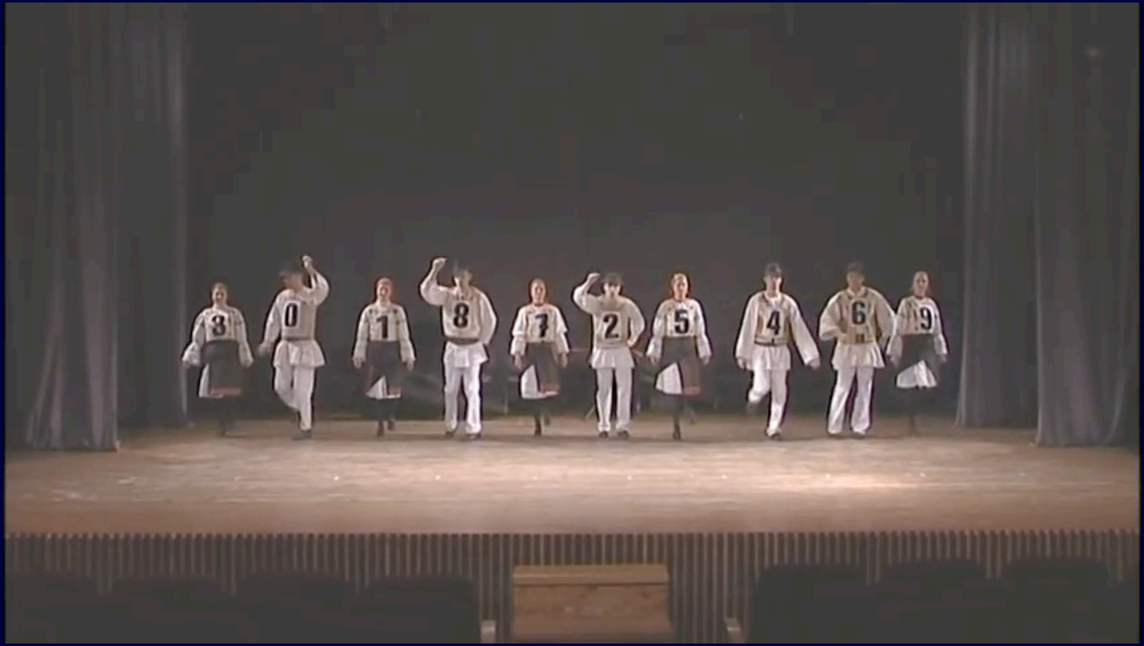
By T. H. Skopinski and Katherine G. Johnson

Langley Research Center
Langley Field, Va.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON September 1960

Standard algorithms

Bubble sort



Created at Sapientia University, Tirgu Mures (Marosvásárhely), Romania. Directed by Kátai Zoltán and Tóth László. In cooperation with "Maros Művészegyüttes", Tirgu Mures (Marosvásárhely), Romania. Choreographer: Füzesi Albert. Video: Lőrinc Lajos, Körmöcki Zoltán. Supported by "Szülőföld Alap", MITIS (NGO) and evoline company.

https://www.youtube.com/watch?v=lyZQPjUT5B4

Bubble sort

3 0 1 8 7 2 5 4 6 9

0 3 1 8 7 2 5 4 6 9

0 1 3 8 7 2 5 4 6 9

0 1 3 8 7 2 5 4 6 9

0 1 3 7 8 2 5 4 6 9

0 1 3 7 2 8 5 4 6 9

0 1 3 7 2 8 5 4 6 9

0 1 3 7 2 5 8 4 6 9

.

.

.

0 1 2 3 5 4 6 7 8 9

0 1 2 3 4 5 6 7 8 9

0 1 2 3 4 5 6 7 8 9

Algorithm complexity

If we had *N* items in a list...

how many steps to sort the list? (time complexity)

how many storage locations to sort the list? (space complexity)

It depends on the data...

If the data are all in order already ... then we go thought the list once (*N* steps)

If the data are in reverse order ...

Algorithm complexity

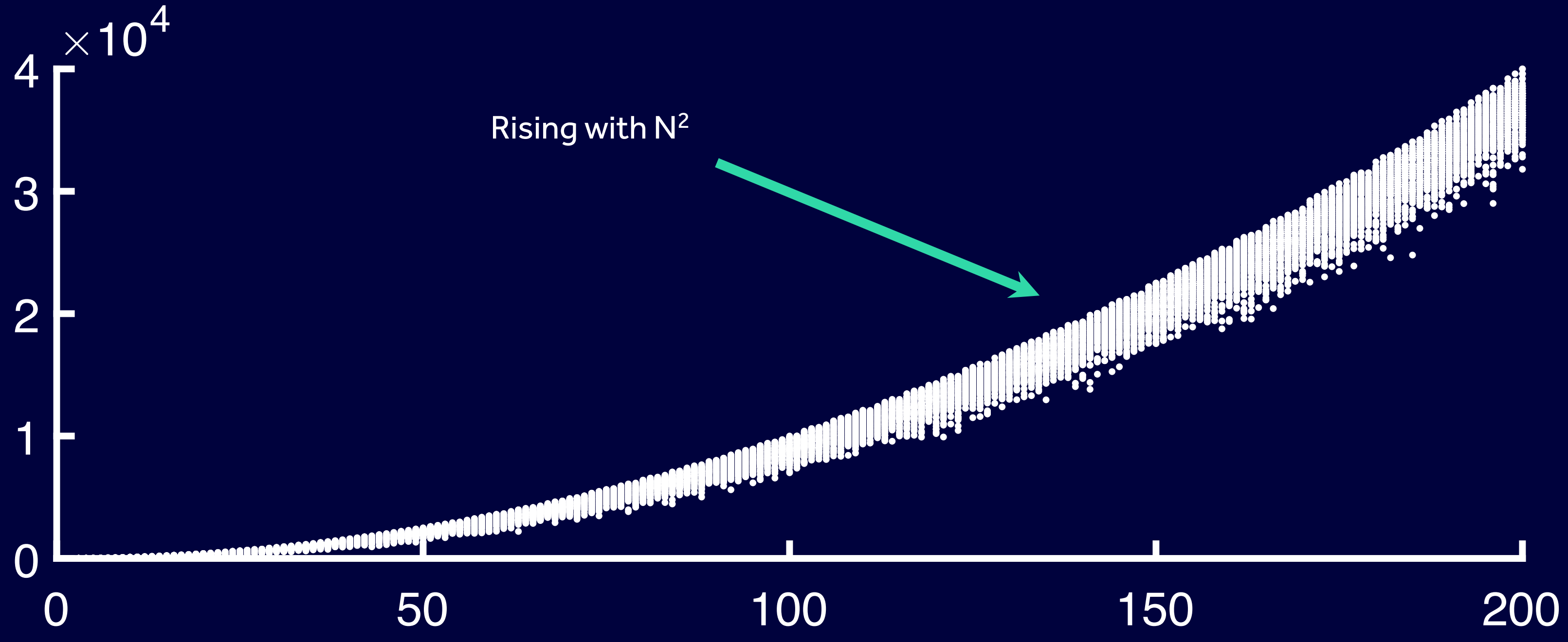
Bubble sort – data in reverse order

Each run along the list takes N steps (because there are N elements)

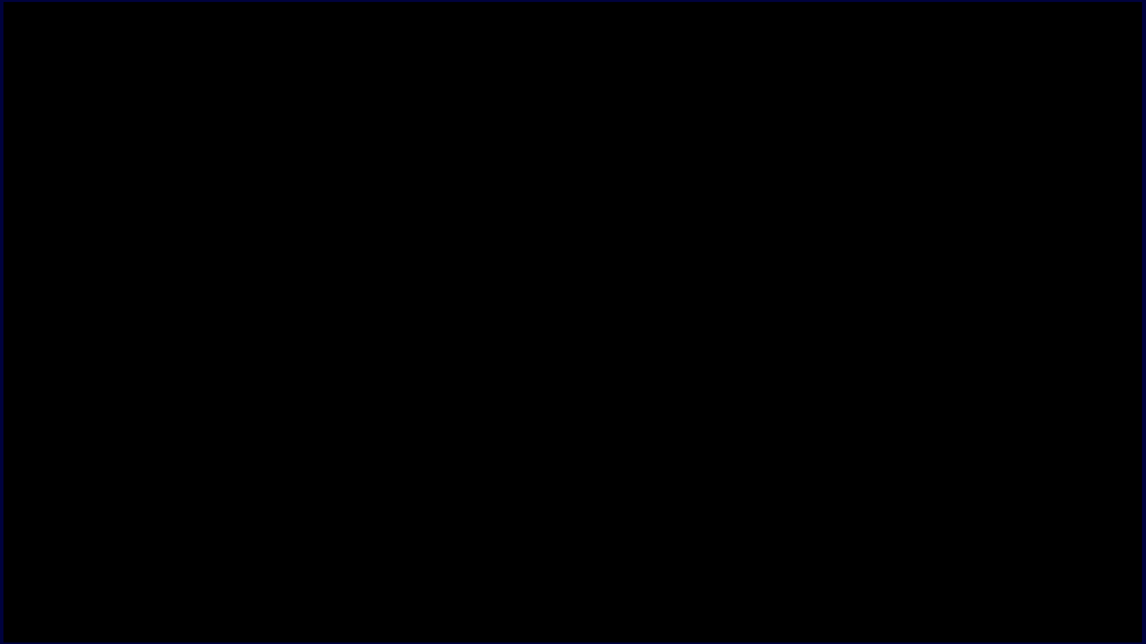
Worst case is we have to move an element N steps along an array.

Hence $N \times N = N^2$. We say the algorithm has complexity $O(N^2)$

Big-O notation



Slight problem…Bubble sort is bad



Created at Sapientia University, Tirgu Mures (Marosvásárhely), Romania. Directed by Kátai Zoltán and Tóth László. In cooperation with "Maros Művészegyüttes", Tirgu Mures (Marosvásárhely), Romania. Choreographer: Füzesi Albert. Video: Lőrinc Lajos, Körmöcki Zoltán. Supported by "Szülőföld Alap", MITIS (NGO) and evoline company.

https://www.youtube.com/watch?v=XaqR3G_NVoo

Complexity – the common orders

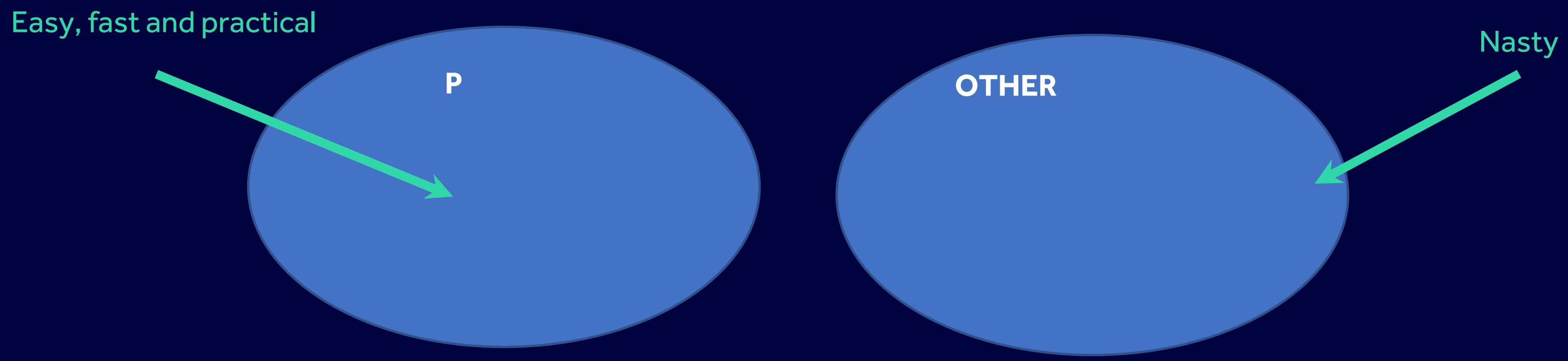
- N
- $\log N$
- $N \log N$
- Polynomial
- Exponential
- Hyper-exponential

Cobham-Edmonds thesis

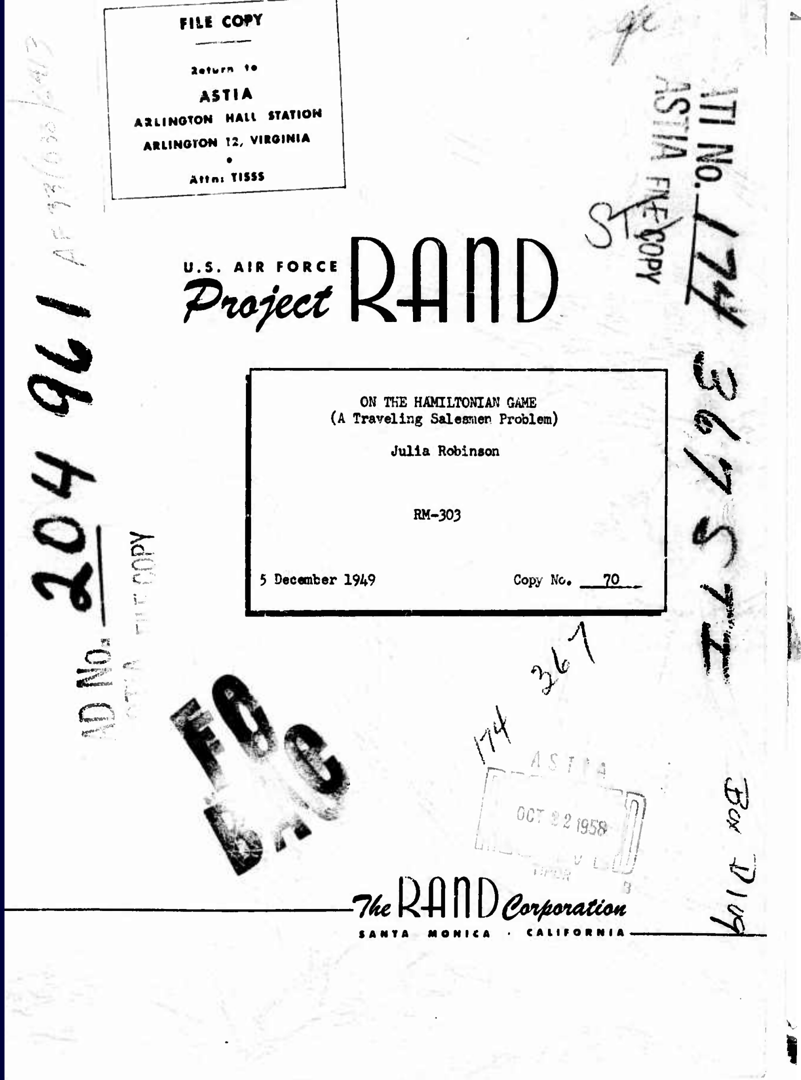
A problem can be feasibly computed if the complexity is polynomial

That is they lie in the complexity class **P**

P is the set of problems decidable in polynomial time



TSP


FILE COPY

Return to

ASTIA

ARLINGTON HALL STATION

ARLINGTON 12, VIRGINIA

Attn: TISSS

U.S. AIR FORCE Project RAND

ON THE HAMILTONIAN GAME

(A Traveling Salesmen Problem)

Julia Robinson

RM-303

5 December 1949 Copy No. 70

AD No. 204 961

ATI No. 174

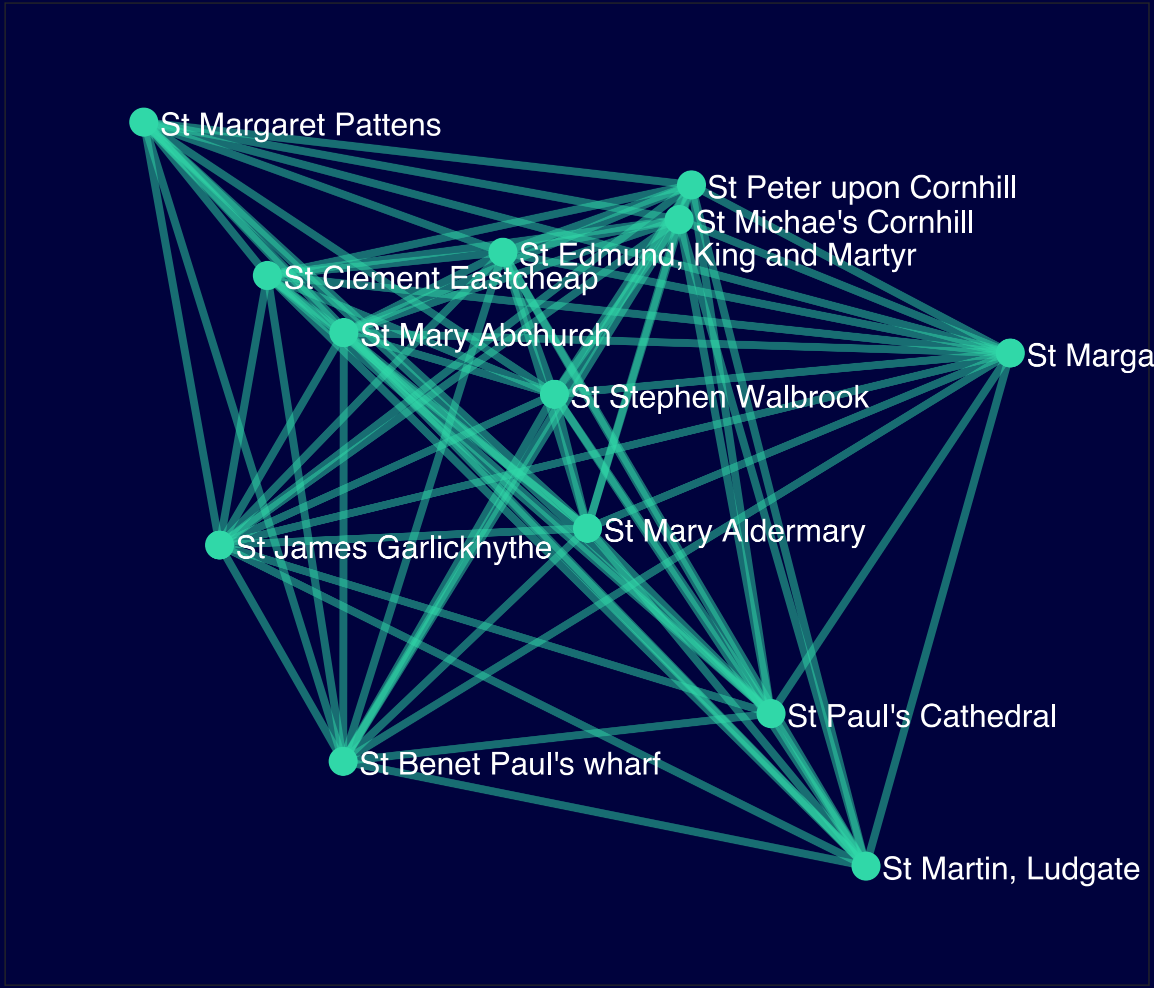
367 57

OCT 22 1958

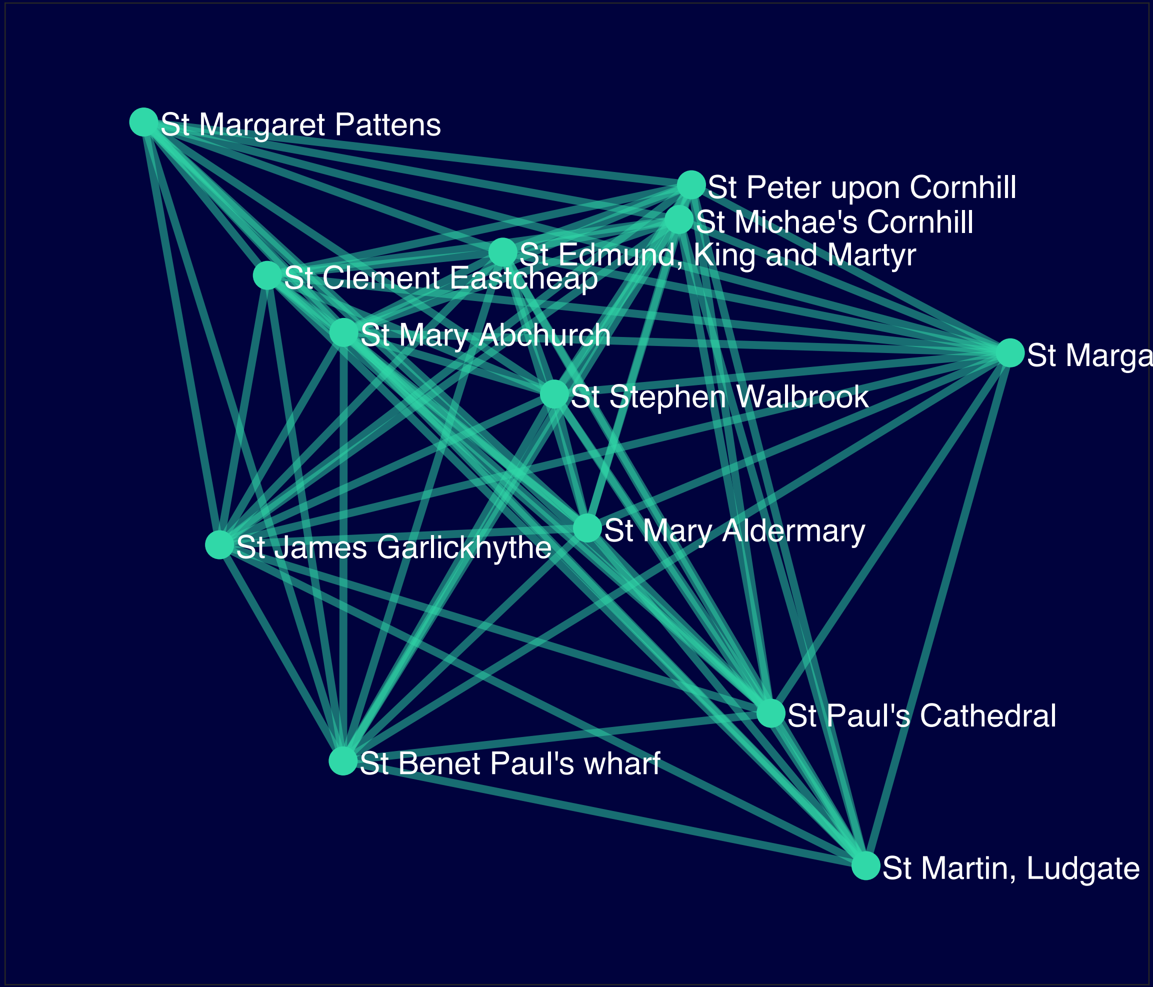
The RAND Corporation

SANTA MONICA · CALIFORNIA

Julia Robinson, *On the Hamiltonian game (The travelling Salesman Problem)*, RAND report RM-303, 5th Dec 1949.



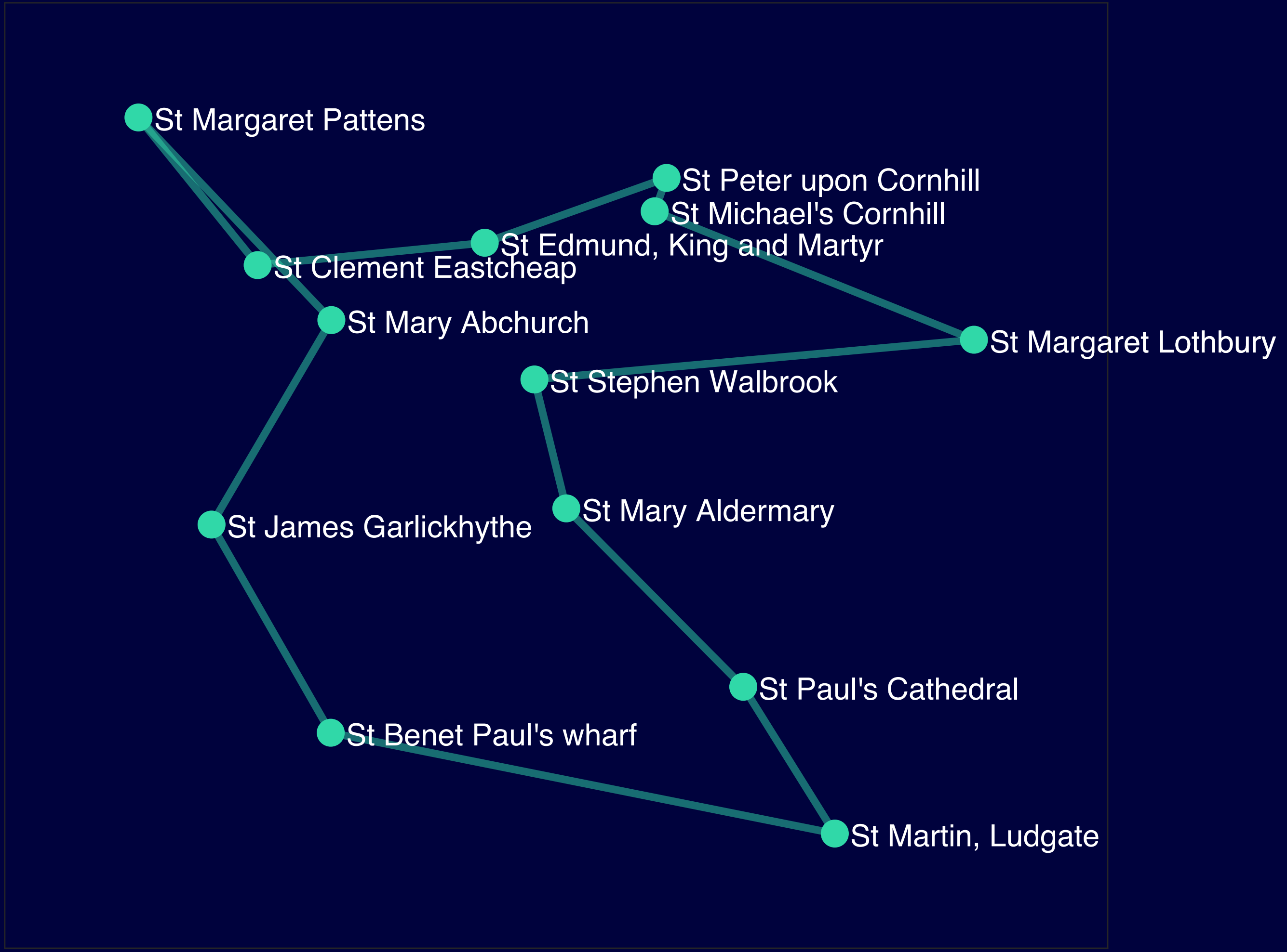
Thirteen churches
Assume Sir Christopher starts at St Paul’s



Thirteen churches
Assume Sir Christopher starts at St Paul's

"Brute force" search...
Check all 12 remaining churches
For each one of the those check 11 remaining..

We have to search...
12 × 11 × 10 × ...2 × 1 = 479,001,600 routes



Optimal route, using symmetrised Google maps walking distances is 4422m

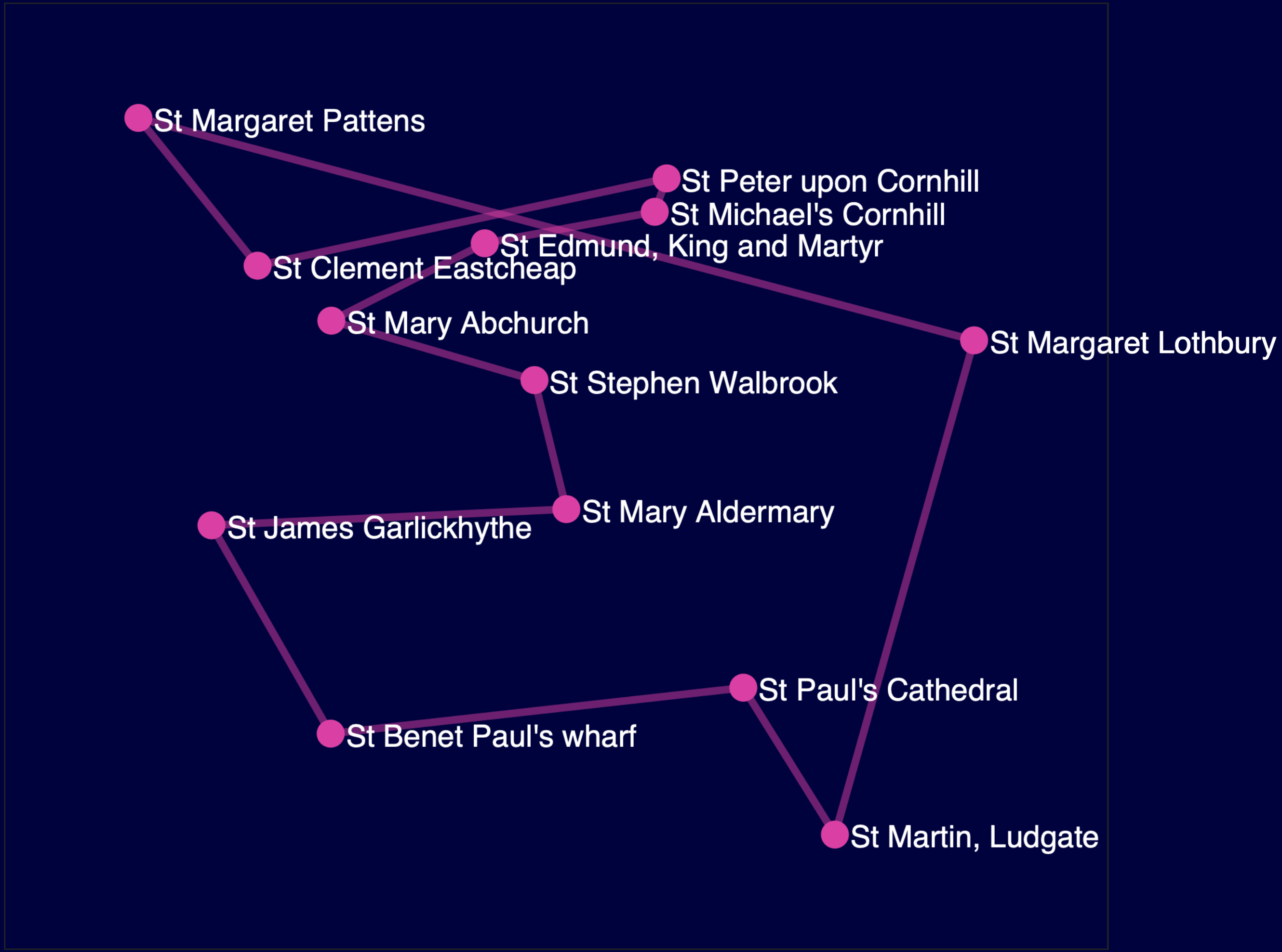
TSP

- But if we wanted to visit all of Wren's London churches?
- There are 52
- My algorithm running on my Mac with 13 churches took around 1ms.
- So 52 churches will take $\frac{52!}{12!}$ ms

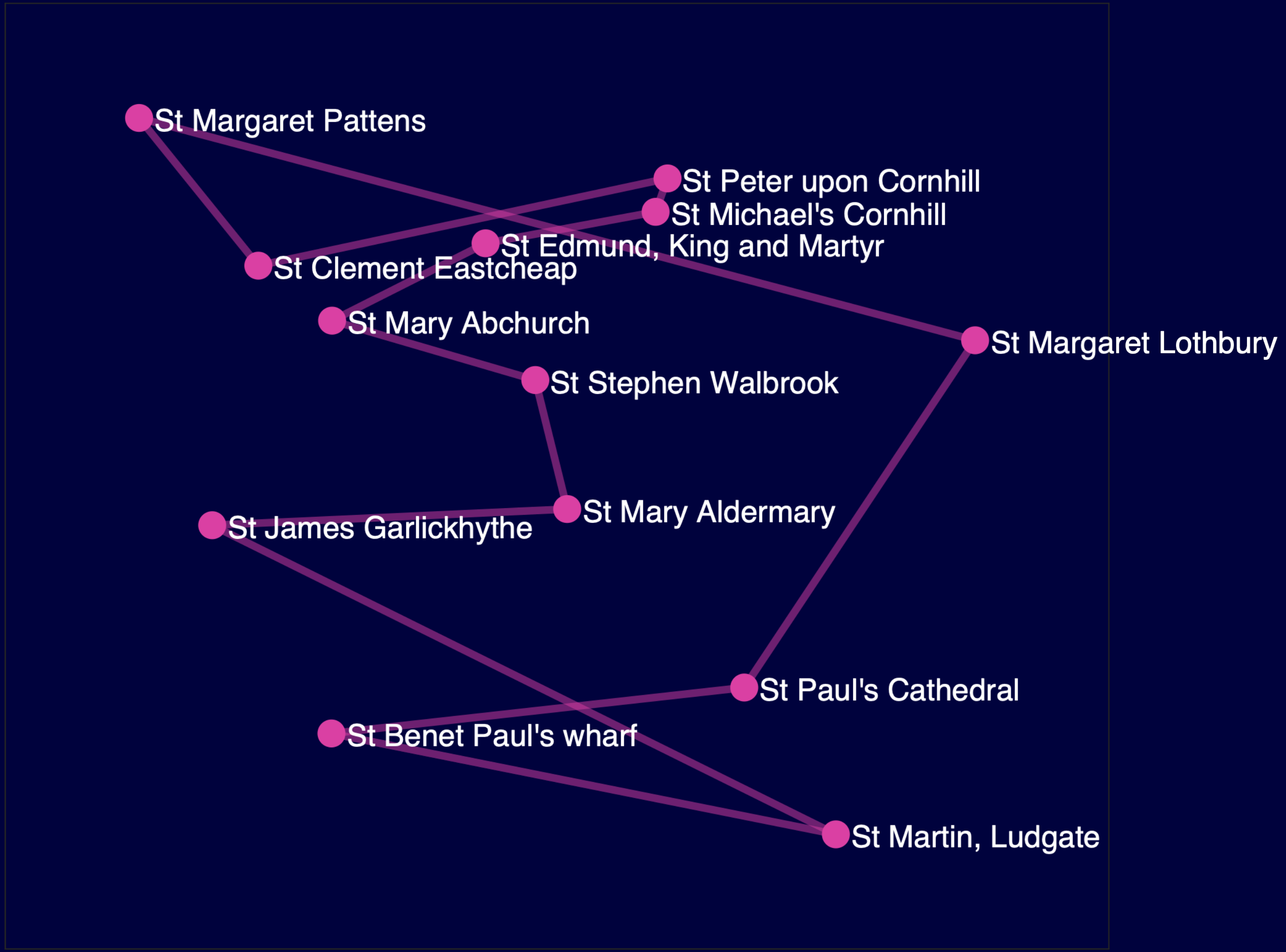
  $= 5.3 \times 10^{48}$ years using brute force!

- So efficient exact algorithms are desired…

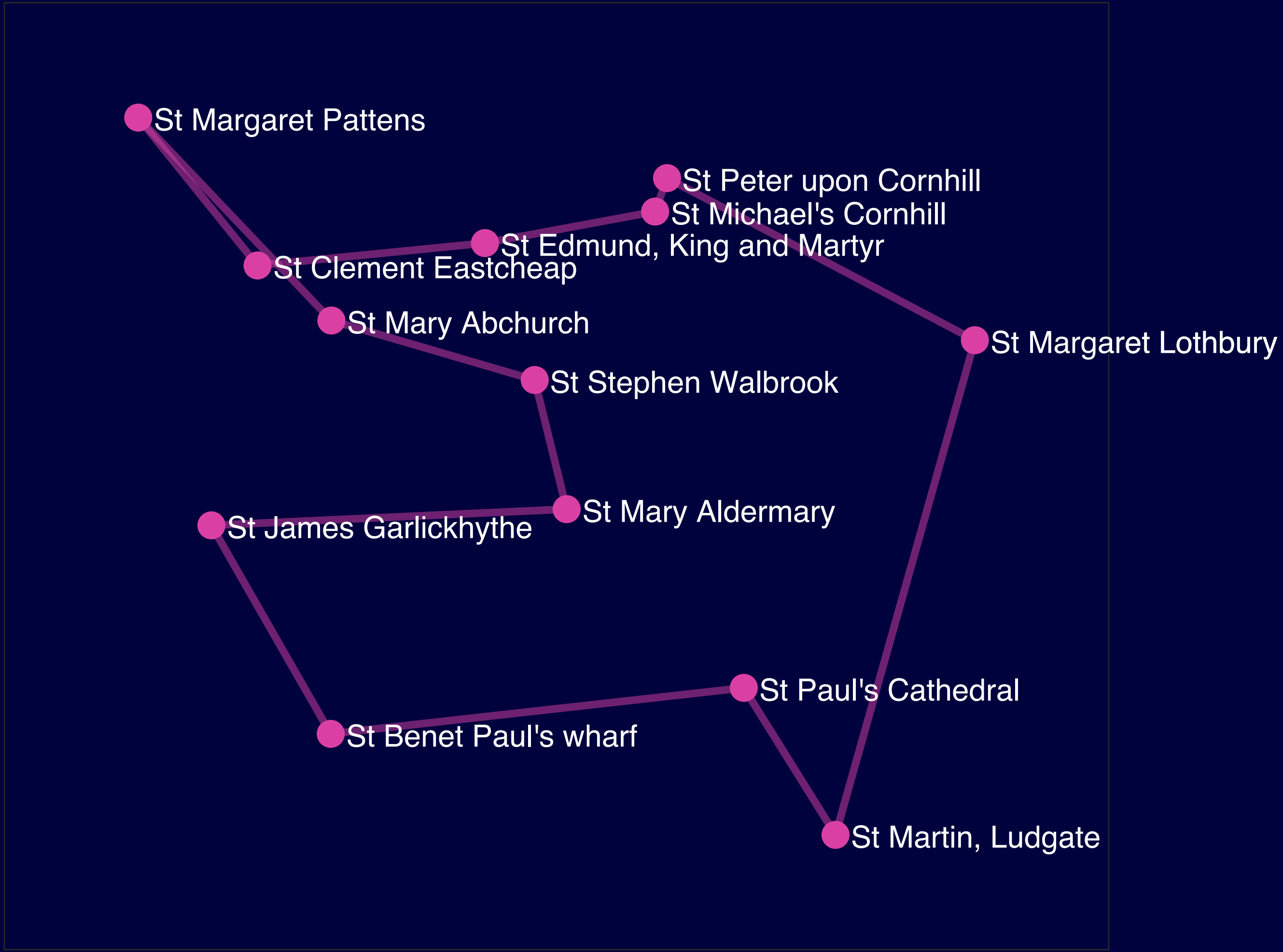
- But none have been found!



Start at St Paul’s
Nearest neighbour
5054 m



Worst NN tour
(St James)
5353 m



Worst NN tour
(St James)
5353 m

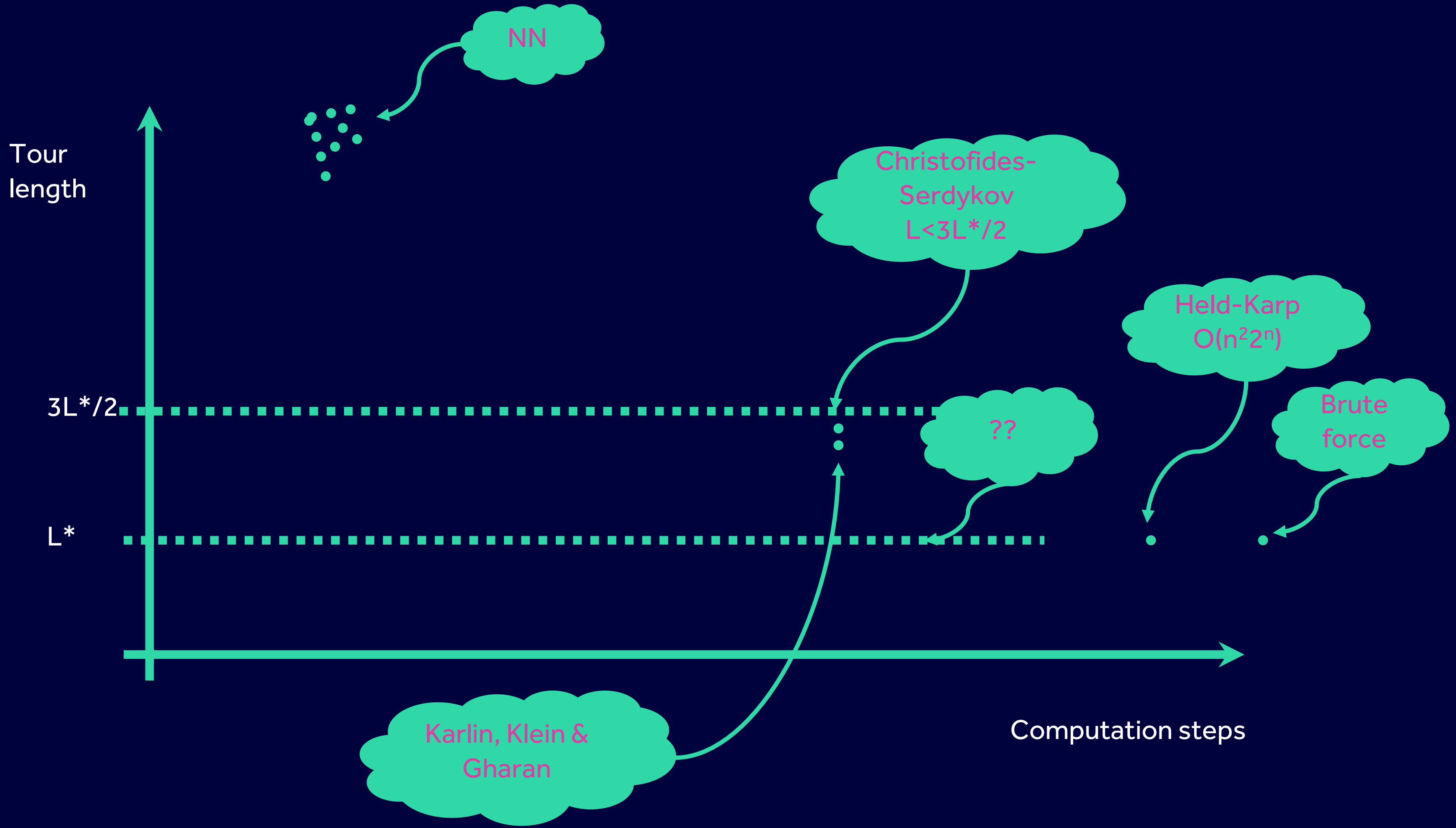
Best NN tour
(St Peter)
4700 m

TSP Bounds

If the length of NN solution is L_{NN} and optimal length is L^* then

$$L^* \leq L_{NN}$$

The NN solution is an *upper bound* on the best solution.



NN
Tour
length
Christofides-
Serdykov
L<3L*/2
Held-Karp
O(n²2ⁿ)
3L*/2
??
Brute
force
L*
Karlin, Klein &
Gharan
Computation steps

Algorithmic structures

- Iteration
- Recursion:

```
factorial(n) = n·factorial(n-1); factorial(1) = 1;


factorial(4)  = 4·3·factorial(3)
              = 4·3·factorial(2)
              = 4·3·2·factorial(1)
              = 4·3·2·1
              = 24
```

Representing algorithms

Pseudocode

Algorithm [edit]

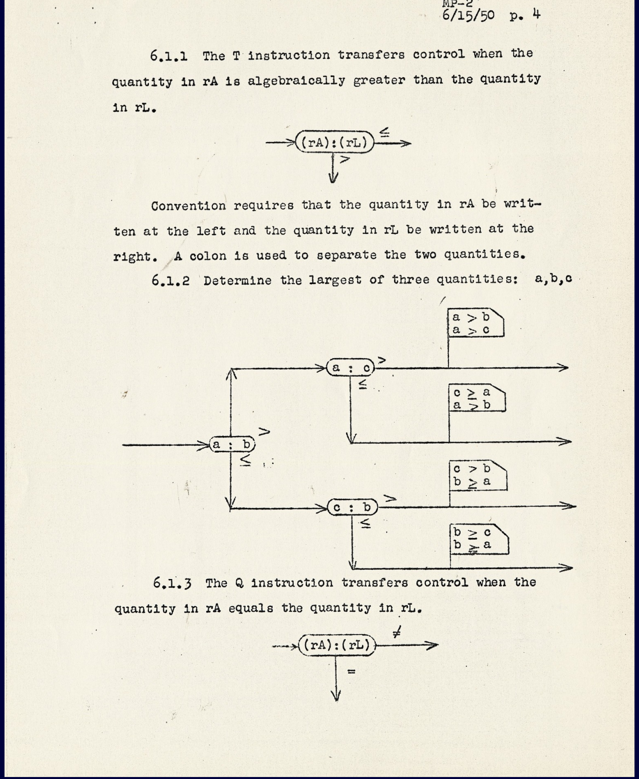
Let $G = (V,w)$ be an instance of the travelling salesman problem. That is, G is a complete graph on the set V of vertices, and the function w assigns a nonnegative real weight to every edge of G. According to the triangle inequality, for every three vertices u, v, and x, it should be the case that $w(uv) + w(vx) \geq w(ux)$.

Then the algorithm can be described in pseudocode as follows.[1]

1. Create a minimum spanning tree T of G.
2. Let O be the set of vertices with odd degree in T. By the handshaking lemma, O has an even number of vertices.
3. Find a minimum-weight perfect matching M in the induced subgraph given by the vertices from O.
4. Combine the edges of M and T to form a connected multigraph H in which each vertex has even degree.
5. Form an Eulerian circuit in H.
6. Make the circuit found in previous step into a Hamiltonian circuit by skipping repeated vertices (*shortcutting*).

Wikipedia article on Christofides-Serdyukov algorithm for solving TSP.

Flowchart


MP-2
6/15/50 p. 4

6.1.1 The T instruction transfers control when the quantity in rA is algebraically greater than the quantity in rL.

Convention requires that the quantity in rA be written at the left and the quantity in rL be written at the right. A colon is used to separate the two quantities.

6.1.2 Determine the largest of three quantities: a,b,c

6.1.3 The Q instruction transfers control when the quantity in rA equals the quantity in rL.

Flowchart attributed to Margery K League from the Grace Murray Hopper Collection at the National Museum of American History. c. 1949

Algorithms more formally

λ-calculus

ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By A. M. TURING.

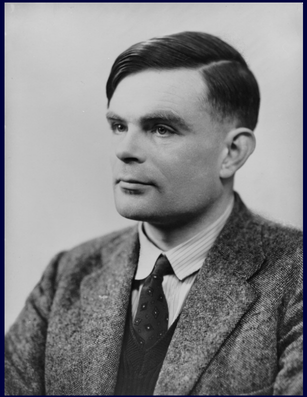
[Received 28 May, 1936.—Read 12 November, 1936.]

The "computable" numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means. Although the subject of this paper is ostensibly the computable *numbers*. it is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable, computable predicates, and so forth. The fundamental problems involved are, however, the same in each case, and I have chosen the computable numbers for explicit treatment as involving the least cumbrous technique. I hope shortly to give an account of the relations of the computable numbers, functions, and so forth to one another. This will include a development of the theory of functions of a real variable expressed in terms of computable numbers. According to my definition, a number is computable if its decimal can be written down by a machine.

In §§ 9, 10 I give some arguments with the intention of showing that the computable numbers include all numbers which could naturally be regarded as computable. In particular, I show that certain large classes of numbers are computable. They include, for instance, the real parts of all algebraic numbers, the real parts of the zeros of the Bessel functions. the numbers π, e, etc. The computable numbers do not, however, include all definable numbers, and an example is given of a definable number which is not computable.

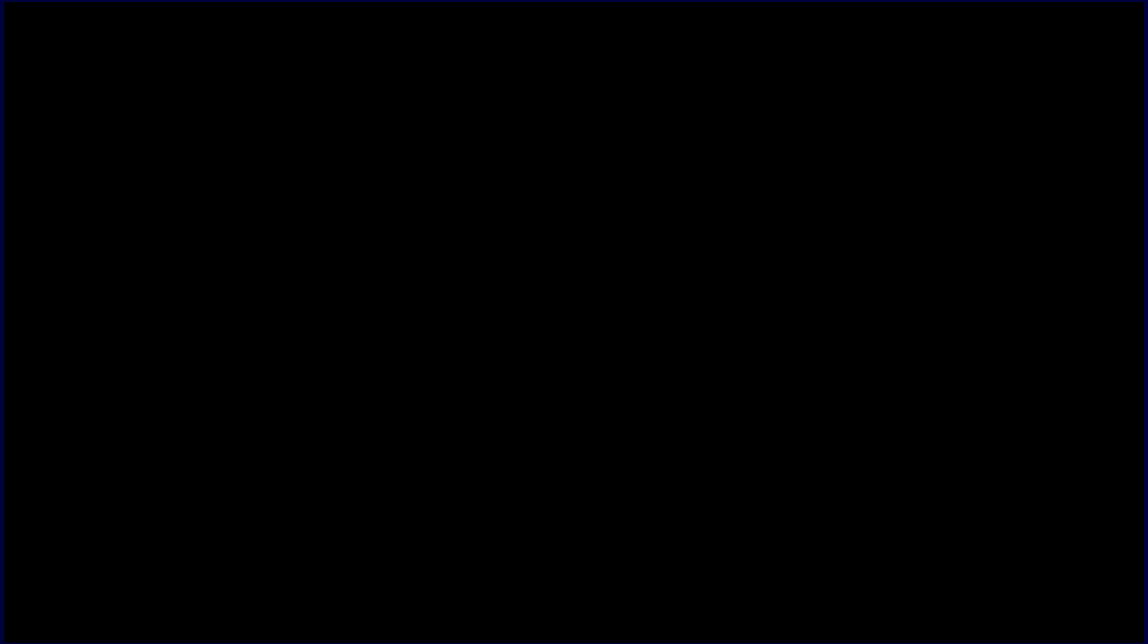
Although the class of computable numbers is so great, and in many ways similar to the class of real numbers, it is nevertheless enumerable. In § 8 I examine certain arguments which would seem to prove the contrary. By the correct application of one of these arguments, conclusions are reached which are superficially similar to those of Gödel†. These results

† Gödel, "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I", *Monatshefte Math. Phys.*, 38 (1931), 173–198.



Turing machines

Turing machine



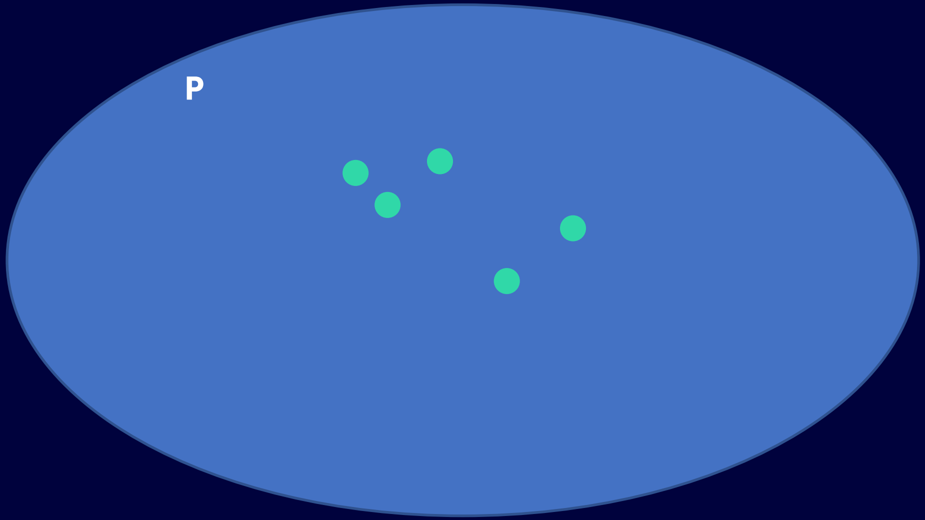
A Turing Machine – Overview [YouTube] [http://aturingmachine.com]

Decision problem

- A problem that has a yes/no answer
- Function problems can be converted into decision problems
  - $z = x + y$ becomes "Does $z = x + y$?"
  - Find the shortest circuit (TSP) becomes "Is there a circuit that is less than some number?"

P

- The set of all decision problems that can be solved by a deterministic Turing machine in a polynomial time



NP

- Is a solution verifiable in polynomial time?

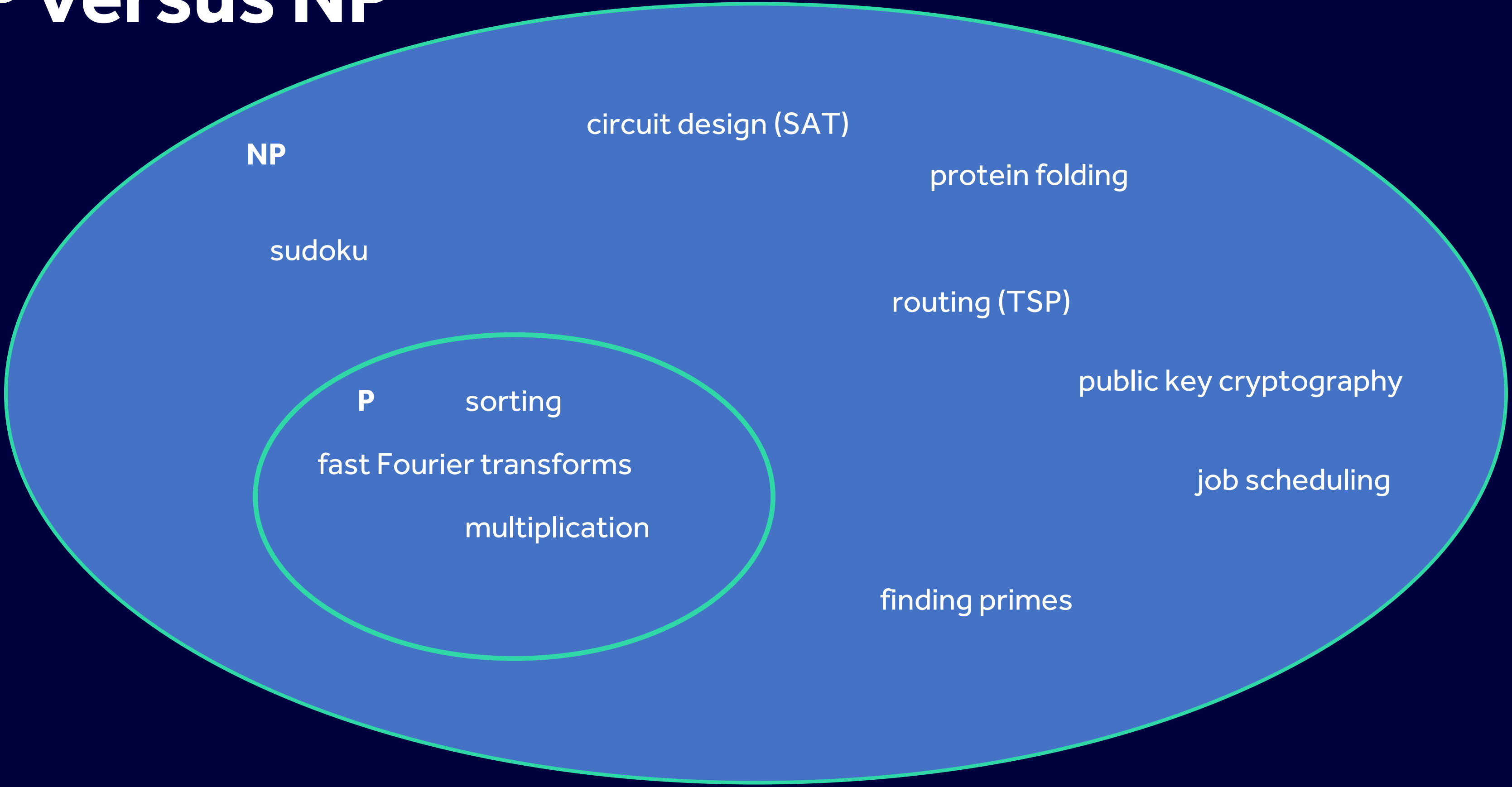
- Sudoku

  Verification is polynomial

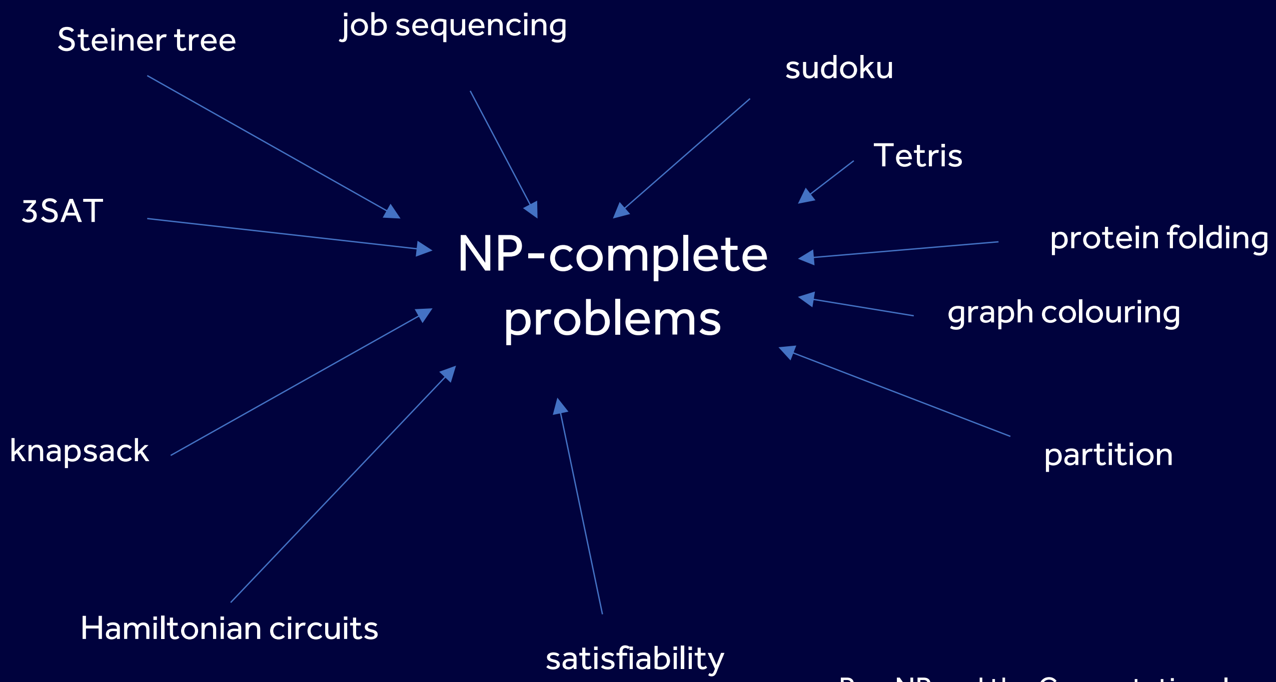
  If we are given a solution then we can verify it in polynomial time

	1	2		3	4	5	6	7
	3	4	5		6	1	8	2
		1		5	8	2		6
		8	6					1
	2				7		5	
		3	7		5		2	8
	8			6		7		
2		7		8	3	6	1	5

P versus NP

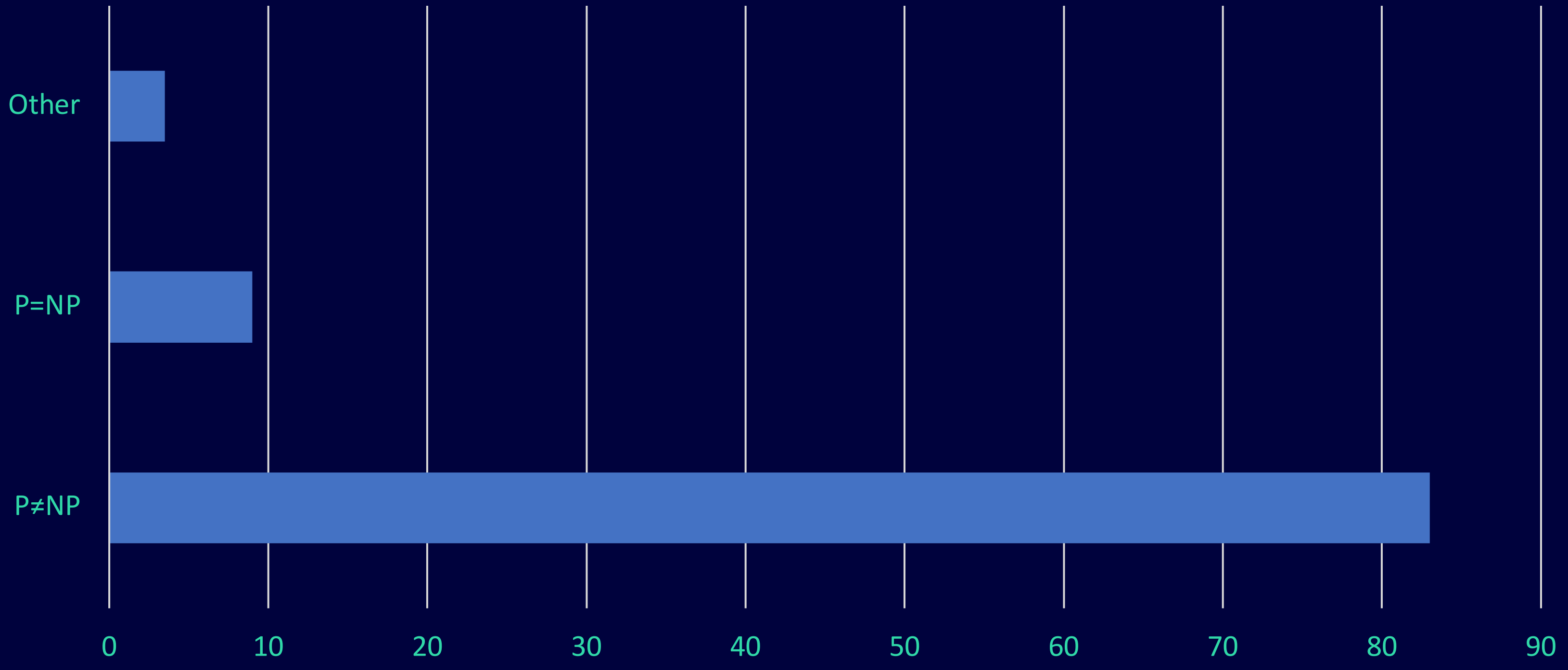

NP
circuit design (SAT)
protein folding
sudoku
routing (TSP)
public key cryptography
job scheduling
finding primes
P
sorting
fast Fourier transforms
multiplication

Many problems are *reducible*



P vs NP and the Computational complexity zoo, hackerdashery, YouTube 2014
https://www.youtube.com/watch?v=YX4OhbAHx3s

What do complexity theorists think?



Algorithm asymmetry

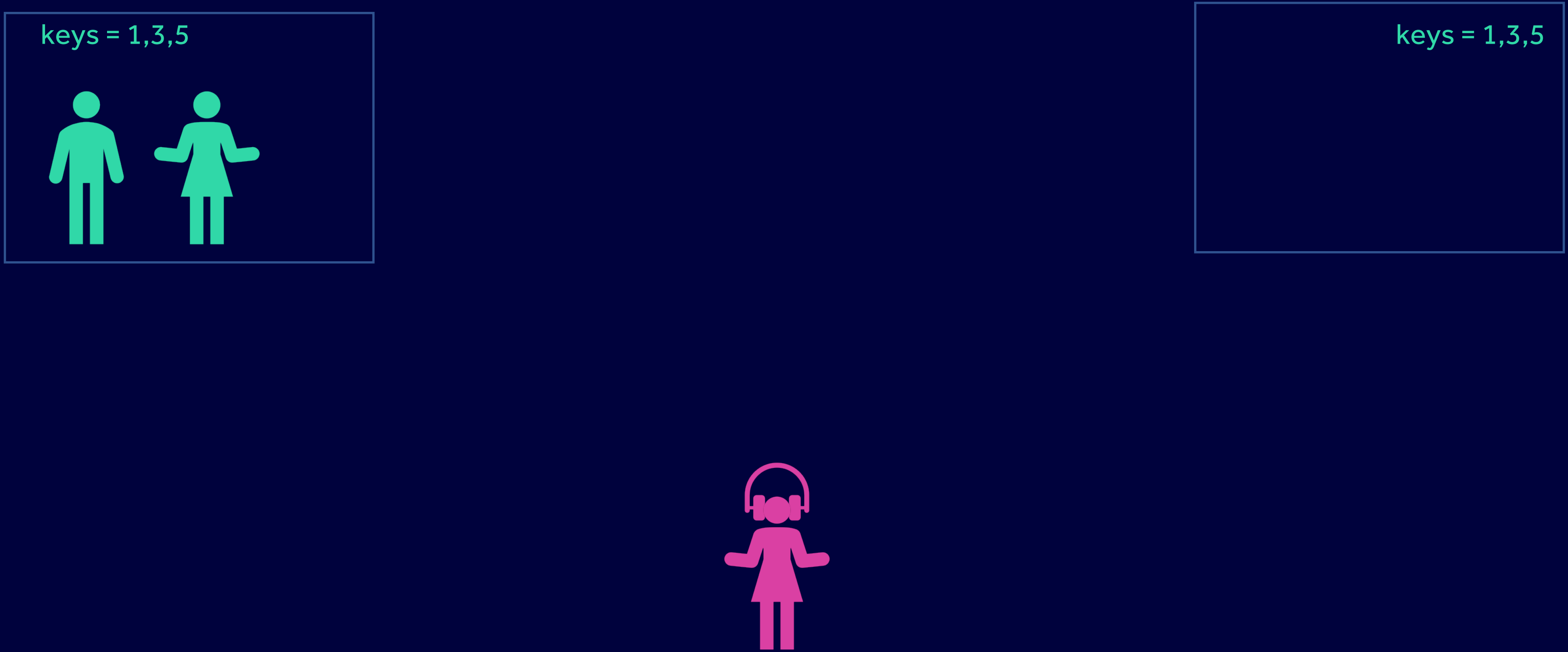
- Algorithms that take ages to compute a solution but for which it is trivial to verify a solution are also useful.

- Subset-sum problem.
  - Given a number, S, can we compute positive numbers, n_i, such that

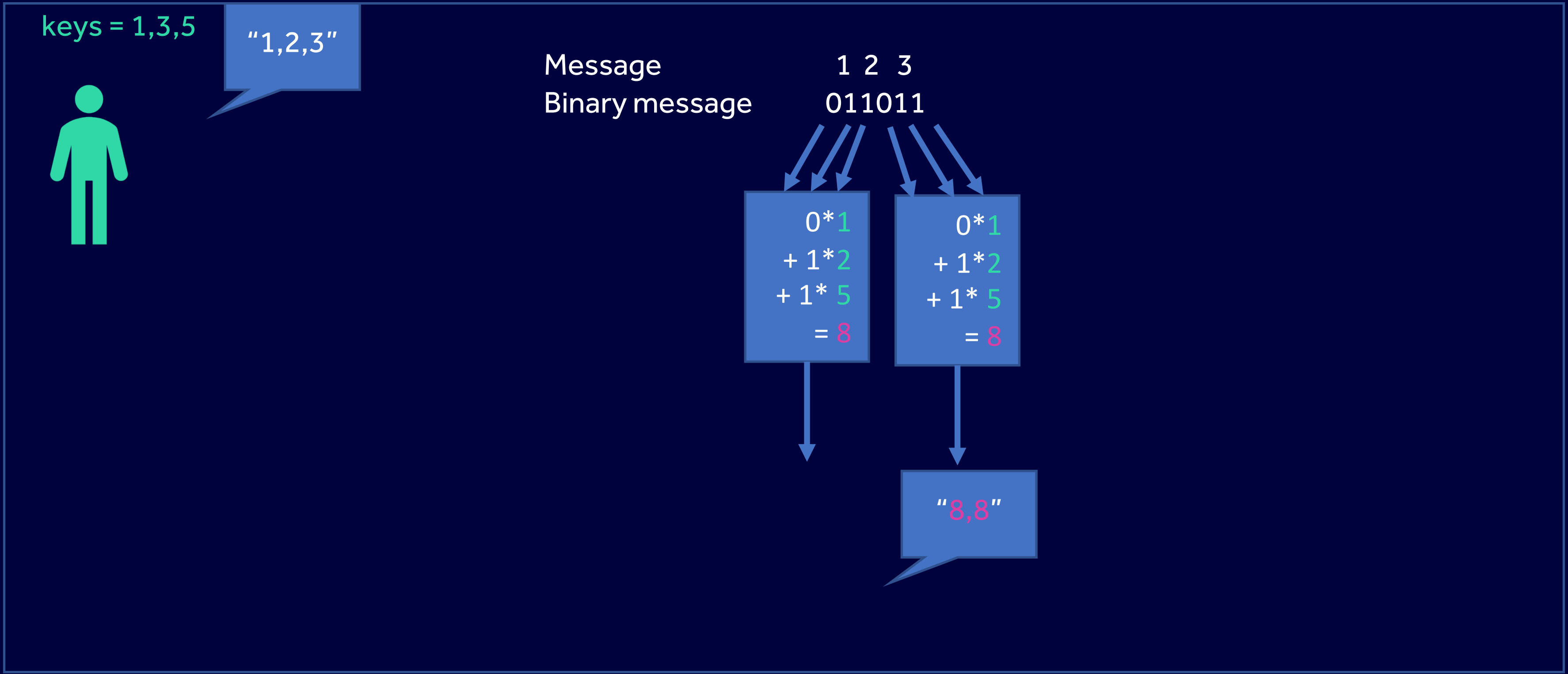
$$S = n_1 + n_2 + \ldots n_\mathrm{m}?$$

  - Obviously verification is trivial – we just add up the numbers and check they make S.

Algorithm asymmetry

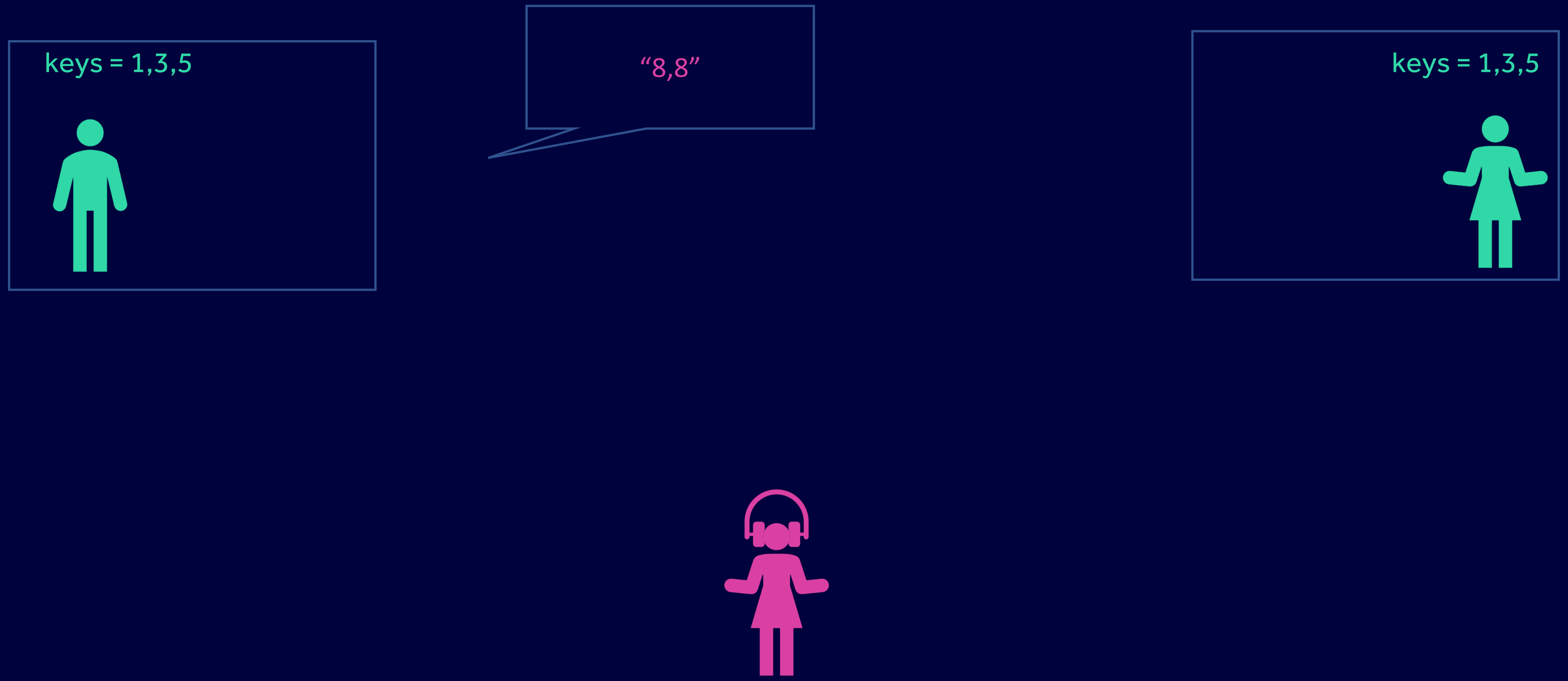


Bob's world



keys = 1,3,5
"1,2,3"
Message 1 2 3
Binary message 011011
0*1
+ 1*2
+ 1* 5
= 8
0*1
+ 1*2
+ 1* 5
= 8
"8,8"

Algorithm asymmetry



Alice's world

keys = 1,3,5

```
decoded = [0 0 0]
key = largest(keys)
i= 3
while (message – key) > 0
        decoded(i) = 1
        message = message – key
        i =i -1
end
```

message	decode
8	[0 0 0]
3	[0 0 1]
0	[0 1 1]

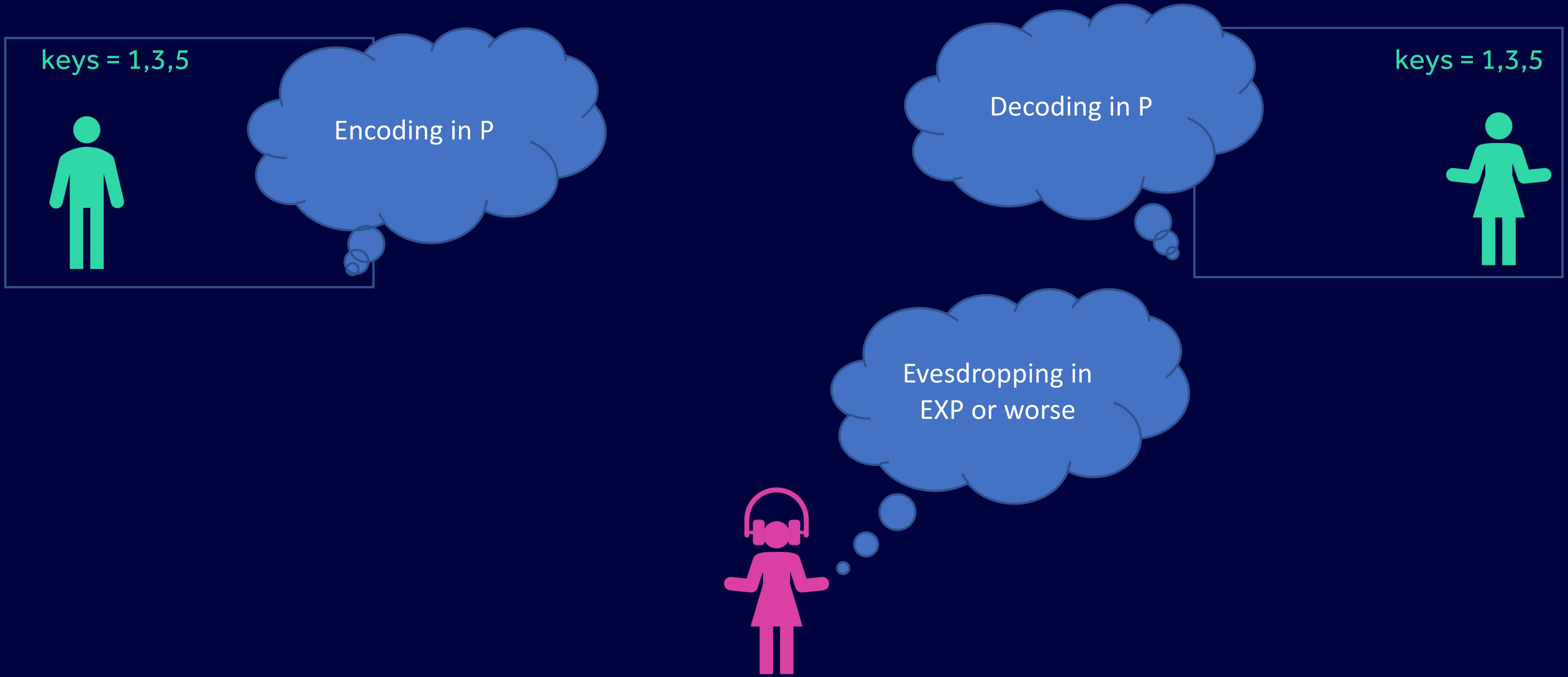
"01 10 11" is "1,2,3"

Eve's world

Assume, via espionage, Eve knows the number of keys and she knows they are in the interval [1,8]

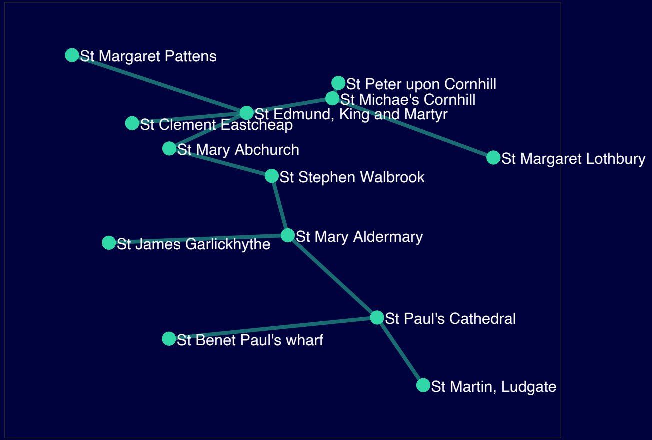
0 0 0					
0 0 1	? ? 8				
0 1 0	? 8 ?				
0 1 1	? 7 1	? 1 7	? 6 2	? 5 3	? 3 5
1 0 0	8 ? ?				
1 0 1	7 ? 1	1 ? 7	6 ? 2	5 ? 3	3 ? 5
1 1 0	7 1 ?	1 7 ?	6 2 ?	5 3 ?	3 5 ?
1 1 1	5 2 1	5 1 2	1 2 5	2 1 5	

Algorithm asymmetry



Algorithms that morph

- Problems that go from easy to hard with minor tweaks
- What if, in TSP, if we temporality allowed backtracking (retracting our steps) at no cost?
- Ah! That is a much easier problem called a Minimal Spanning Tree (MST)
- Solve the MST problem and then edit the solution to avoid revisits.

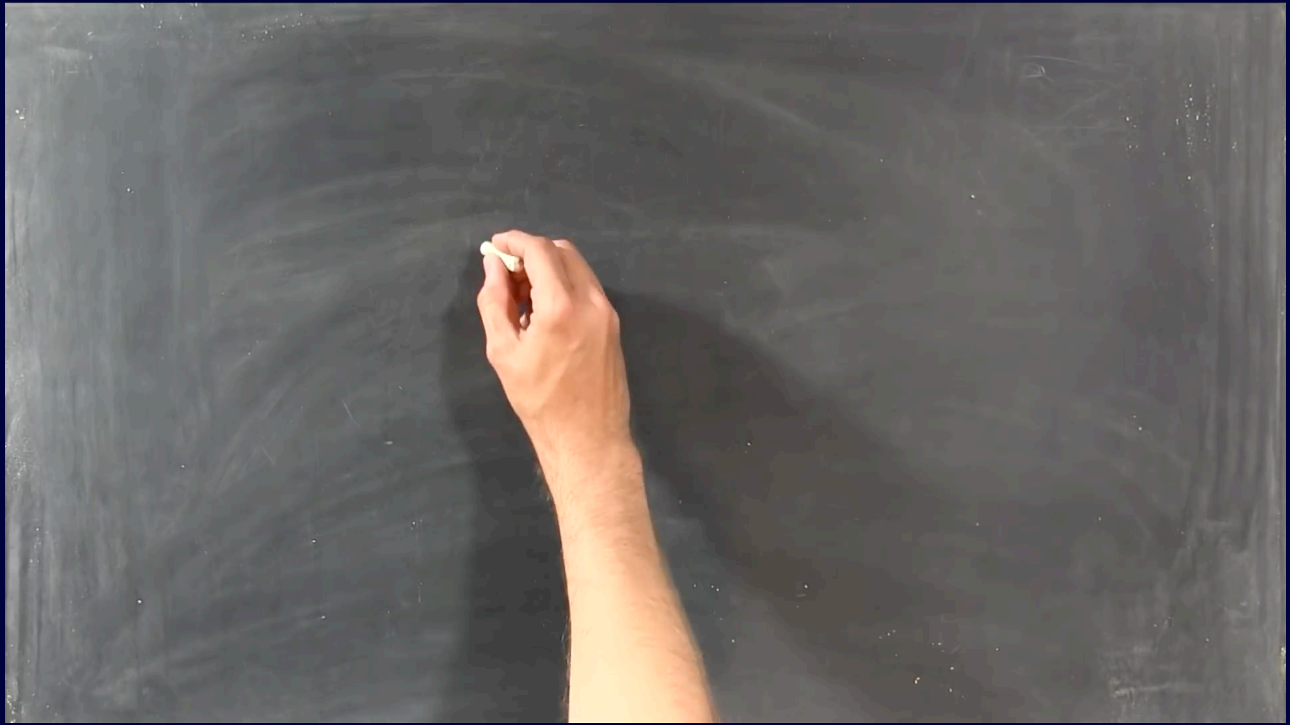


Where next for complexity?

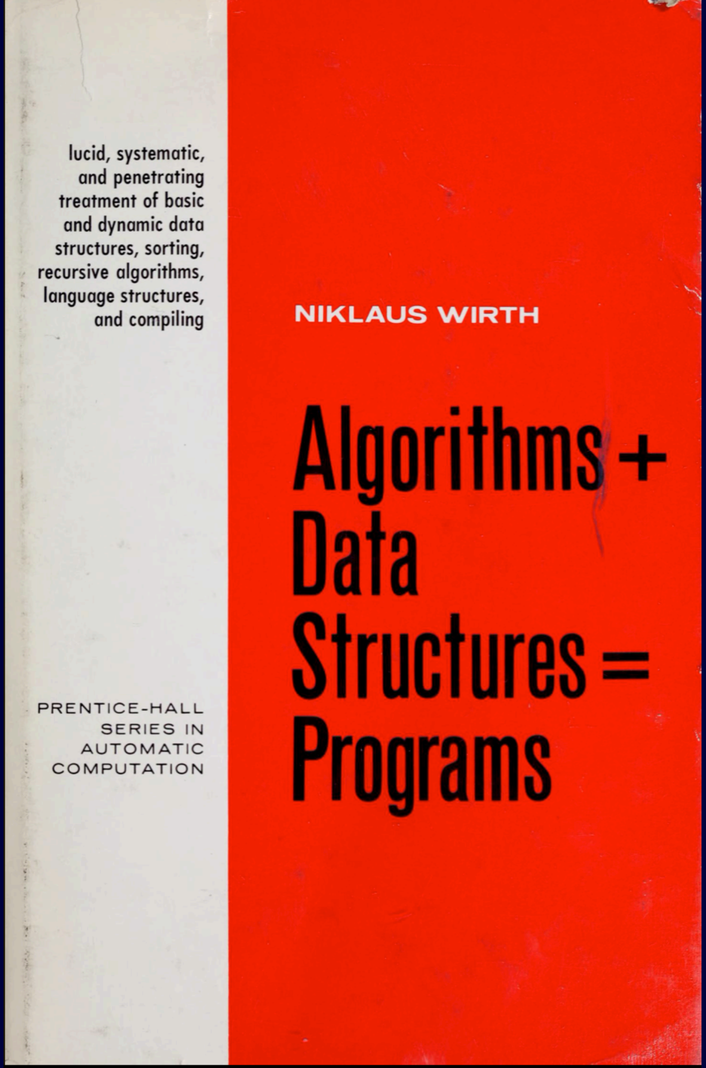
- Proving that P = NP is tricky because
  - Proving something is itself an NP problem
- There are plenty of complexity classes
  - 417 according to Scott Aaronson!
  - Includes quantum classes
  - Problems that are not decision problems
  - Problems that are more difficult than polynomial

Important point – just because a decision problem is in a difficult complexity class; it does not mean we can do nothing – Amazon drivers still deliver!

The complexity zoo



P vs NP and the Computational Complexity Zoo, Hackerdashery, YouTube Aug 2014
https://www.youtube.com/watch?v=YX40hbAHx3s



Algorithms (this lecture)

Data structures (24th Nov 2020 18:00)

Programs (2nd Feb 2021 18:00)

Computers (9th March 2021 18:00)

Networks (20th April 2021 18:00)

Security (25th May 2021 18:00)