The NP-completeness proofs we looked at so far are all standard and from decades ago.

Today:

Some interesting/more recent NP-completeness results and some open problems on NP-completeness.

This lecture is enrichment.

Almost all “natural” problems in NP are known to be in P or NP-complete.

Exceptions open in Garey and Johnston’s famous book on NP-completeness, 1979

Linear Programming	in P, 1980	
Primality	in P, 2002	
Graph Isomorphism	still open	not known to be in coNP either

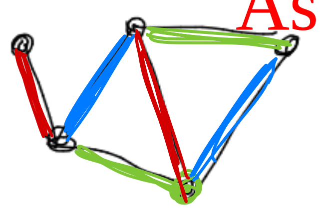
Given two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are they the same graph up to relabelling?

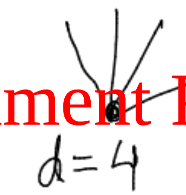
Graph isomorphism has many applications, e.g. in chemistry.

A surprising NP-completeness result.

Edge colouring.
Colour the edges of a graph such that incident edges have different colours.

Example





degree d ⇒ need d colours

Δ = max degree ⇒ need Δ colours

Theorem. [Vizing, 1964] The minimum number of colours is Δ or $\Delta + 1$.
W https://en.wikipedia.org/wiki/Vizing%27s_theorem

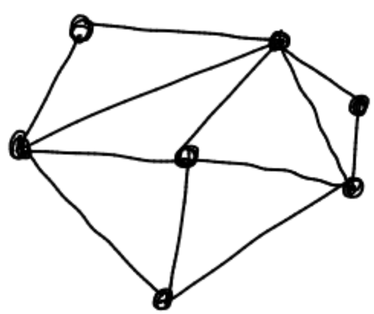
But deciding if a graph can be coloured with Δ colours is NP-complete.

Interesting, more recent NP-completeness proofs

1. Nintendo games are NP-hard, Erik Demaine et al., 2012. https://arxiv.org/pdf/1203.1895v1.pdf

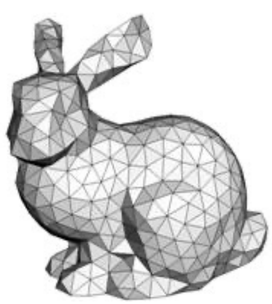
   reduction from 3-SAT

2. Minimum weight triangulation. Mulzer and Rote, 2008. https://dl.acm.org/doi/10.1145/1346330.1346336



Given points in the plane and number k,
is there a *triangulation* with sum of edge lengths ≤ k ?

Curiously, this problem is not known to be in NP
because verification involves checking if a sum of
square roots is ≤ k,
and no one knows how to do that in polynomial time
(round off errors in square roots).



triangulations are very useful

Interesting, more recent NP-completeness proofs

3. Rubik's cube. Demaine, Eisenstat, Rudoy, 2018.

https://drops.dagstuhl.de/opus/volltexte/2018/8533/pdf/LIPIcs-STACS-2018-24.pdf

Given an n × n × n Rubik's cube
(messed up) can it be solved in ≤ k moves?

"move" = rotating a slice

Note that for a 3 × 3 × 3 Rubik's cube, the question of a polynomial time algorithm is moot because there is a finite set of possibilities.

Finite but HUGE! 43 x 10^{18}

And the maximum number of moves is 20, but only proved in 2013.

fascinating description of how this was done: https://cube20.org

Some open problems. Are the following NP-complete?

1. Factoring. W https://en.wikipedia.org/wiki/Integer_factorization

Given numbers n, k, does n have a factor ≤ k ?

e.g. n = 35, k=4.

NO. $35 = 5 \times 7$ prime factorization

Factoring ∈ NP ∩ coNP

obvious

by above example, since
testing primes is in P

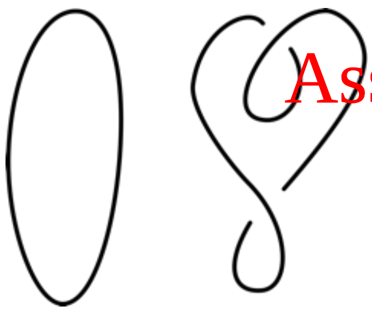
The public-key cryptosystem RSA depends on the difficulty of factoring. But factoring can be done in polynomial time on a quantum computer [Peter Shor, 1994]

“post-quantum cryptography” W https://en.wikipedia.org/wiki/Post-quantum_cryptography
— current possibilities are based on lattice problems.

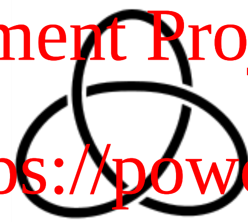
Some open problems. Are the following NP-complete?

2. The unknot problem. W https://en.wikipedia.org/wiki/Unknotting_problem

Given a knot diagram, does it represent the unknot?



two diagrams of the unknot



the trefoil knot

is this the unknot?

1999. The unknot problem is in NP.

2016. The unknot problem is in co-NP.

Summary of Lecture 21.5

there is on-going work on NP-completeness and some great open questions

Next:

What to do when the problem you want to solve is NP-complete.