Algorithms, complexity and P vs NP

Can creativity be automated?

Slides by Avi Wigderson + Bernard Chazelle (with some extras)

SURVEY

Finding an efficient method to solve SuDoku puzzles is:

		8	6					
							6	
			4	8			2	3
		5		9				8
	4	9				2	1	
2				4		7		
3	6			2	9			
	1							
					5	1		

1: A waste of time
2: A decent way to pass some time
3: A fundamental problem of science and math

Algorithms

Function: input → output
Addition: x,y → x+y

12345
+ 6789
→ addition algorithm → 19134

input → output

ALGORITHM (intuitive def):
Step-by-step, simple procedure, computing a function on *all* inputs

---

Evaluating how good (how efficient) an algorithm is

How does the number of basic steps of an algorithm increase with the data size (input length) ?

input

Rubik's cube




2 3 4 5 ...

Sudoku

		8	6					
							6	
			4	8			2	3
		5		9				8
	4	9				2	1	
2				4		7		
3	6			2	9			
	1							
					5	1		

3

1			2	3	4			12		6				7	
	8				7			3			9	10	6	11	
	12			10			1		13		11			14	
3			15	2			14				9			12	
13				8			10		12	2		1	15		
	11	7	6				16				15			5	13
			10		5	15			4		8			11	
16			5	9	12			1						8	
	2						13			12	5	8			3
	13			15		3			14	8		16			
5	8			1				2				13	9	15	
		12	4		6	16		13			7				5
	3			12				6			4	11			16
	7			16		5		14			1			2	
11	1	15	9			13			2				14		
	14				11		2			13	3	5			12

4

Sudoku

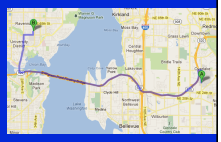
	y				b		a	c	x	n			h			t		f	l				d	e
	t		s	u	j	h	v				d	q	c			o	k		b	n			a	w	p
w	h	e	m	a	n		l	u	k	p		r	y		s	x	d	q	c	o	j		l	b
b		j		p	s			t				l	m		v	n	g	h	a	q		r	x	y
x	o	l	d		l	p			r	e			f		u	j	w	y	m		h		s	c
	w	q	u	j			l		e		x	b	o	m	a			n	h	k	c	s		
n	c				w	x	u	s	f		q		l			e	m	k	v				j	a
a	l		x	f	c	l			m		v	k	w		q			j		d	g		b	h
s			v		h	k	p	o	b	u	f	j	n				t		d	l	m		r	q
	b	d		m	f	v			j		h	p			o	g	y	w			t		u	
y	p		e	l	a	m		v	h	o	b		x	l	t	s	q	u	w	g	r	c	d	k
	q	g	j		e		s	r		h	c				f	k			x		y	l	a	o
		u	t	k		n	o		l		r	m	q	y		b	a	v	j			l	p	h
	x	r		w	p		y	k	l		l	e	j					m		t	q	v		u
	s		n	b	q	c		g	w	k	a	u	t	p	y		o		r	x		j	m	
j	n	s	q	v	x	y	h		u	t	p	o	g	l	m		f	d			w	l	k	r
u		w	b	t	l	e	r	p	o	m		c	d	f	k	v					s	q		
d			h		m	s	c	f		q	j		k	n	g	w		b		l	v	u		e
			o	e	d	l	k	n	q		w		u		j	a	l			h		b	p	m
l	k				v	j	t	w		a	s	h					u	r	q	c	d	f		n
			g	d	y	r	w			c		l	l		n	p	v	a	f	e			q	
	v	x	p	q		t	b			d	n	f			w			g		s	a	h	y	l
l		k	w	c	g	q	x	h					a	u	l	d	e		s			m	f	v
		a	y	r		d	f	e	n	x	k		s	h			b		u		p			
q	l		f	s			m	l	v			w			h		x	t	y				c	d

5 .....

The Algorithm Incredible gems

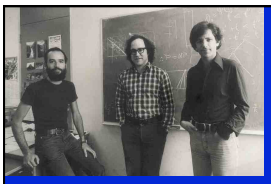
Google maps

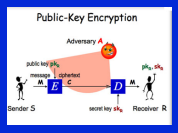
Shortest paths



FFT

Audio processing
Image processing
Tomography, MRI





RSA

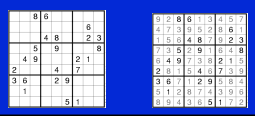
encryption
e-commerce

The class P

All problems having an efficient algorithm to *find* solutions

Are all practically interesting problems in P?

Three problems

	Input	Output	Complexity
Factoring integers	1541	23 x67	
	$2^{67}-1$	193,707,721 × 761,838,257,287	$\leq 2^{\sqrt{n}}$
Proving theorems	n+"Riemann Hypothesis"	n symbol proof	$\leq 2^n$
Solving Sudoku			$\leq n^n$

Verification

$2^{67}-1$ = 193707721 x 761838257287

n+Poincare Conjecture n+Fermat's "Theorem"

n = 200 pages

	Input	Output	Complexity
Factoring integers	1541	23 x67	
	$2^{67}-1$	??	$\leq 2^{\sqrt{n}}$
Proving theorems	n+"Riemann Hypothesis"	n symbol proof	$\leq 2^n$
Solving Sudoku			$\leq n^n$

What is common to all 3 problems?

- Best current algorithms exponential

-Easy verification of given solutions !!!

The class NP

All problems

- whose solutions can be written down in polynomial space
- having efficient verification algorithms for given solutions

P versus NP

P: Problems for which solutions can be efficiently *found*

NP: Problems for which solutions can be efficiently *verified*

Fact: $P \subseteq NP$ [finding implies verification]

Conjecture: $P \neq NP$ [finding is much harder than verification]

"P=NP?" is a central question of math, science & technology !!!

what is in NP?

Mathematician: Given a statement, *find* a proof

Scientist: Given data on some phenomena, *find* a theory explaining it.

Engineer: Given constraints (size,weight,energy) *find* a design (bridge, medicine, phone)

In many intellectual challenges, *verifying* that we found a good solution is an easy task ! (if not, we probably wouldn't start looking)

If P=NP, these have fast, automatic *finder*

How do we tackle P vs. NP?

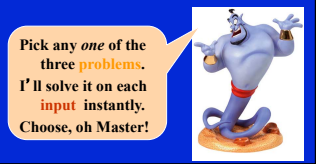
Break RSA, ruin E-commerce

Fame & glory $6M from CLAY

Take out the fun of Doing these puzzles

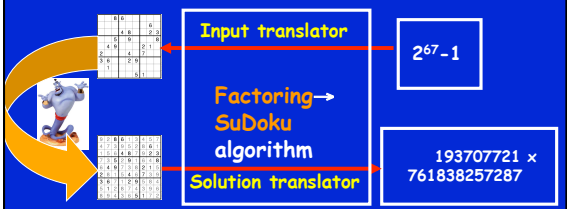
	Input	Output	Complexity
Factoring integers	1541	23 x67	
	$2^{67}-1$	??	$\leq 2^{\sqrt{n}}$
Proving theorems	n+"Riemann Hypothesis"	n symbol proof	$\leq 2^n$
Solving SuDoku			$\leq n^n$

Let's choose the SuDoku solver



The power of SuDoku I

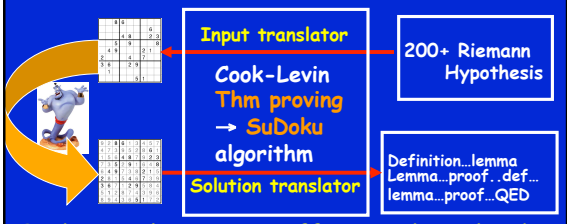
Using SuDoku solver for Integer factoring



Both translators are efficient algorithms!

The power of SuDoku II

Using SuDoku solver for Theorem proving



Both translators are efficient algorithms!

"Reduction"

"If you give me a place to stand, I will move the earth."
– Archimedes (~ 250BC)

"If you give me a polynomial-time algorithm for Sudoku, I will give you a polynomial-time algorithm for every NP problem." --- Cook, Levin (1971)

"Every NP problem is a Sudoku problem in disguise."

Universality: NP-completeness

SuDoku solver can solve any NP problem

1971: NP-complete problems exist!

SAT is NP-complete: There is a "reduction" from any NP problem to SAT

NP-complete problems abound!

1972: 21 problems in logic, optimization, algebra

Today: ~3000 problems in all sciences, *equivalent*

P=NP iff SuDoku has an efficient algorithm

Universality: NP-completeness

NP-complete problems:

If one is easy, then all are!

If one is hard, then all are!

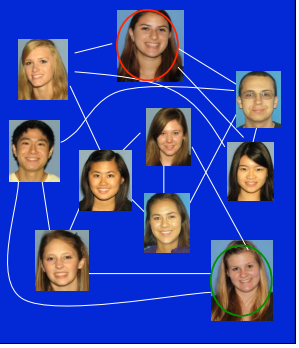
SuDoku, NP-complete

Thm proving: NP-complete

Integer factoring: we don't know

Rumor mill problem

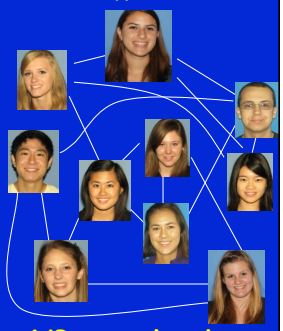
- Social network
- Each node represents a student
- Two nodes connected by edge if those students are friends
- Melissa starts a rumor
- Will it reach Jordyn?
- How does the running time of our algorithm depend on the size of the input?



In P

CLIQUE Problem

- Social network
- Each node represents a student
- Two nodes connected by edge if those students are friends
- In this social network, is there a clique of k or more people?
- CLIQUE: Group of students, every pair of whom are friends
- What is a good algorithm for detecting the biggest clique?
- How does efficiency depend on network size and desired clique size?



NP-complete!

Map Coloring

Input: planar map M

4-COL: is M 4-colorable?

YES!

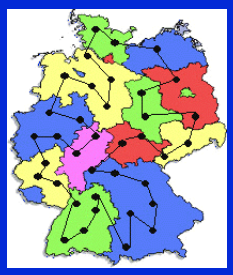
3-COL: is M 3-colorable?

Give me an algorithm.

NP-complete!

Traveling Salesman Problem (aka UPS Truck problem)

- Input: *n* points and all pairwise inter-point distances, and a distance *k*
- Decide: is there a path that visits all the points ("salesman tour") whose total length is at most *k*?



- NP-complete!