Satisfiability of circuits and equations in algebras

Piotr Kawałek

TU Wien

15 Dec 2023

ERC Synergy Grant POCOCOP (GA 101071674)

$$E=mc^2$$

$$IP = PSPACE$$

$$x+y = y+x$$

$$F=ma$$

$$\Delta = b^2-4ac$$

Equations

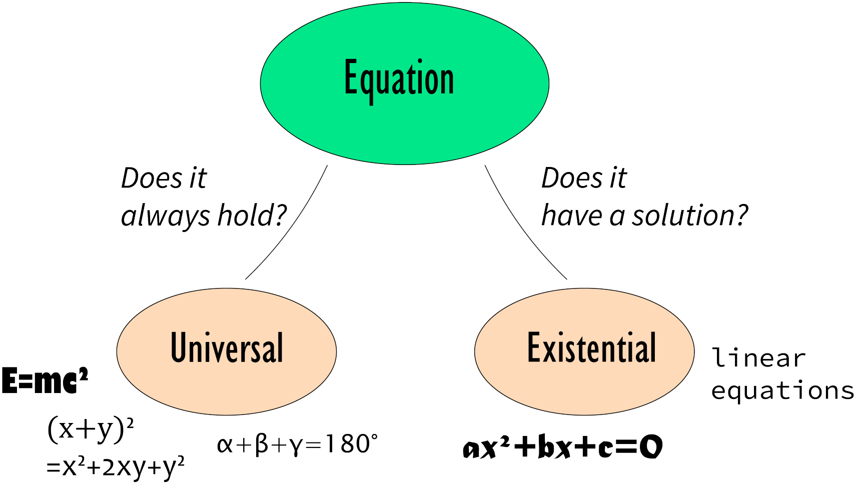
$$e^{\iota\pi}+1 = 0$$

$$pV = nRt$$

$$\mathcal{PT}+\mathcal{DP} = ♥$$

$$\alpha+\beta+\gamma=180°$$

$$a^2+b^2=c^2$$


Equation
Does it
always hold?
Does it
have a solution?
Universal
Existential
E=mc²
(x+y)²
=x²+2xy+y²
α+β+γ=180°
linear
equations
ax²+bx+c=0

Some examples

Example 1. Systems of linear equations over $(\mathbb{Q}, +, -, 0, 1)$

$$\begin{cases} 3x + 2y = 1 \\ 2x + 3y = 2 \end{cases}$$

Method:
Gaussian Elimination Algorithm
Time: $O(n^3)$ fast

The problem is in P.

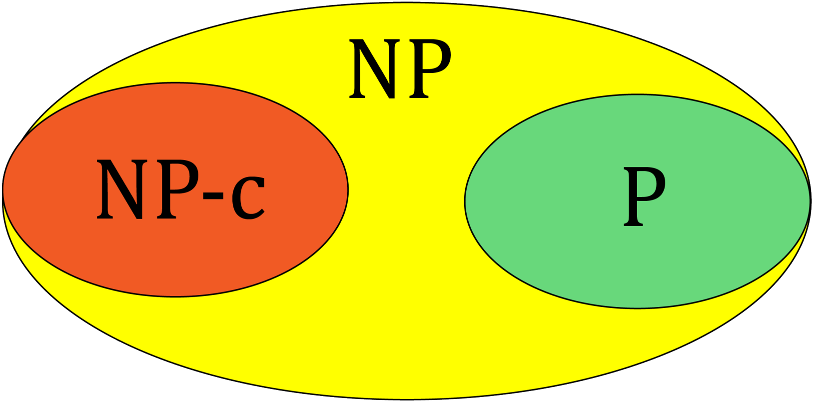
Example 2. Single equation in finite fields $(\mathbb{F}_q, +, -, \cdot, 0, 1)$

$$(x - y) \cdot z + t = 3$$

Method:
Brute-Force
Time: $O(q^n)$ very slow

The problem is NP-complete

P $\neq$ NP



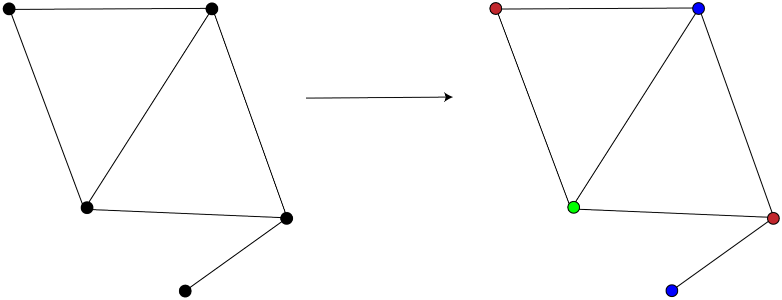
$P \neq NP$

Strategy to prove that some algorithmic problem **B** is hard to solve:

1. Take an NP-complete problem **A**.
2. Show that you can solve **A** quickly by using **B**.
3. Then **B** is also NP-complete, so **B** is hard to solve assuming P $\neq$ NP.

It is hard to paint

k-coloring



k-coloring is NP-complete for $k \geqslant 3$.

3-SAT

Task: given a Boolean formula in 3-CNF form, find a satisfying assignment to the variables.

Example: $(x_1 \vee x_2 \vee \sim x_3) \wedge (\sim x_2 \vee \sim x_3) \wedge (x_1 \vee x_3)$.

Solution: $x_1 = 1$, $x_2 = 0$, $x_3 = 0$.

3-SAT is NP-complete

Some examples

Example 1. Systems of linear equations over $(\mathbb{Q}, +, -, 0, 1)$

$$\begin{cases} 3x + 2y = 1 \\ 2x + 3y = 2 \end{cases}$$

Method:
Gaussian Elimination Algorithm
Time: $O(n^3)$ fast

The problem is in P.

Example 2. Single equation in finite fields $(\mathbb{F}_q, +, -, \cdot, 0, 1)$

$$(x - y) \cdot z + t = 3$$

Method:
Brute-Force
Time: $O(q^n)$ very slow

The problem is NP-complete

Single equation in finite fields

Field: $(\mathbb{F}_q, +, -, \cdot, 0, 1)$, $q \geqslant 3$

q-coloring instance: graph with vertices $V = [n] = \{1, \ldots, n\}$ and edges $e^{(1)}, \ldots, e^{(l)} \in V^2$ (let E denote set of all edges)

Looking for: a coloring $c : V \longmapsto [q]$ such that $c(e_1^{(i)}) \neq c(e_2^{(i)})$. for all $0 \leqslant i \leqslant l$.

Encoding as an equation: For each vertex i create variable c_i.

$$\prod_{(i,j) \in E} (c_i - c_j)^{q-1} = 1$$

So if we could solve equations in the field $\mathbb{F}_q$ quickly we could solve q-coloring quickly! **Then, solving equations in $\mathbb{F}_q$ is NP-complete.**

X'th Hilbert's problem

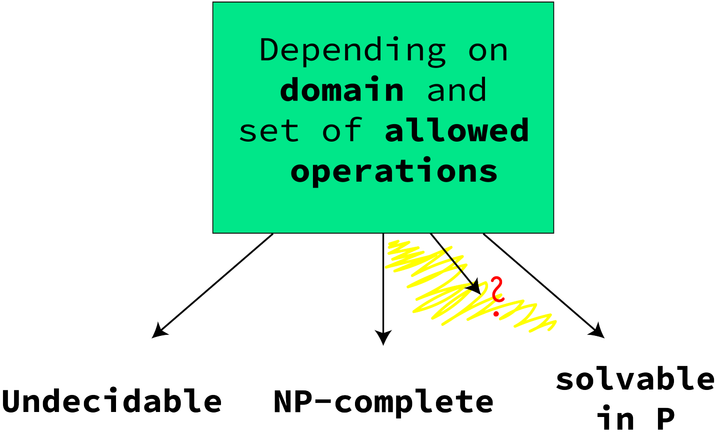
Example 3. Diophantine equations over $(\mathbb{Z}, +, -, \cdot, 0, 1)$

$$a^5 + b^5 = c^5$$

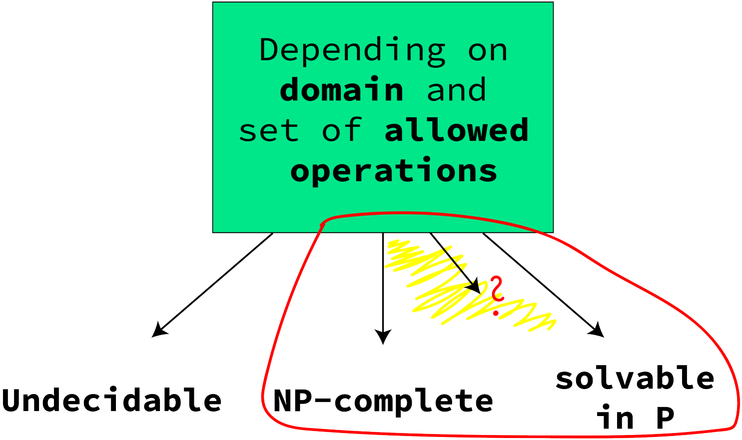
Method:
No general method
Time: ∞ (no algorithmic solution)

The problem is undecidable.
(Matiyasevich 1970)

Some cases


Depending on **domain** and set of **allowed operations**
?
Undecidable
NP-complete
solvable in P

Some cases



Depending on **domain** and set of **allowed operations**
Undecidable
NP-complete
solvable in P
¿

PolSat - definition

SysPolSat(A)

$\mathbf{A} = (A, \mathcal{F})$
A - **finite** domain
$\mathcal{F}$ - list of allowed operations

Problem

IN: $(\mathbf{p}_1, \mathbf{q}_1), \ldots, (\mathbf{p}_m, \mathbf{q}_m)$

OUT: $\exists_{\overline{x}} \ \&_i \ \mathbf{p}_i(\overline{x}) = \mathbf{q}_i(\overline{x})$

PolSat(A)

$\mathbf{A} = (A, \mathcal{F})$
A - **finite** domain
$\mathcal{F}$ - list of allowed operations

Problem

IN: $\mathbf{p}, \mathbf{q}$

OUT: $\exists_{\overline{x}} \ \mathbf{p}(\overline{x}) = \mathbf{q}(\overline{x})$

As the domain is finite, at least we have a brute-force search algorithm.

What are the hopes?

Goal: Describe finite algebraic structures $\mathbf{A} = (A, \mathcal{F})$ for which $\textsc{SysPolSat}(\mathbf{A}) \,/\, \textsc{PolSat}(\mathbf{A})$ has a polynomial-time solution.

Surprise: For $\textsc{SysPolSat}(\mathbf{A})$ we already have such (complicated) description! It is due to the recent results of Zhuk (2017) and Bulatov (2017) about CSPs.

Dychotomy: $\textsc{SysPolSat}(\mathbf{A})$ is either in P (has a fast algorithmic solutions) or is NP-complete (we can reduce k-coloring to it). So the only reason for hardness here is NP-completness.

Question: Is it the same for $\textsc{PolSat}$?

The problem we are left with...

$\textsc{PolSat}(\mathbf{A})$

$\mathbf{A} = (A, \mathcal{F})$

A - **finite** domain

$\mathcal{F}$ - list of allowed operations

Problem

IN: $\mathbf{p}, \mathbf{q}$

OUT: $\exists_{\overline{x}}\ \mathbf{p}(\overline{x}) = \mathbf{q}(\overline{x})$

Let's talk about groups

Signature: $\mathbf{G} = (G, \cdot, ^{-1})$

Equations:

1. $x \cdot y \cdot x^{-1} \cdot y^{-1} = 1$,
2. $x \cdot y = g \cdot y \cdot x$, where $g \in G, g \neq 1$ is a constant
3. $x \cdot y^{-1} \cdot g \cdot z^{-1} \cdot x = h \cdot x^{-1} \cdot y \cdot x$, where $g, h \in G$ are constants, and x, y, z are variables.

Some simple cases

When **G** is abelian it decomposes into a direct product

$$\mathbf{G} \equiv \prod_{i=1}^{k} \mathbb{Z}_{p_i^{k_i}}$$

So we can solve equation separately on each coordinate $\mathbb{Z}_{p_i^{k_i}}$.
But for $\mathbf{G} = \mathbb{Z}_{p^k}$ we rewrite the equation to the form:

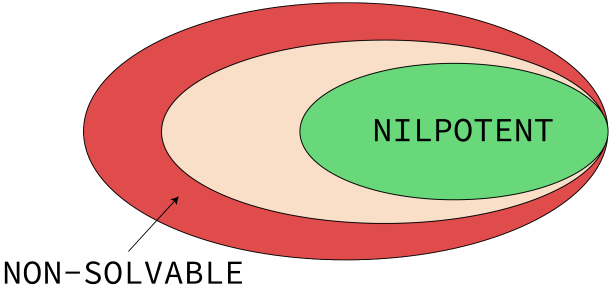
$$\sum_{i=1}^{n} \alpha_i x_i = c$$

And now there is a solution iff $\gcd(\alpha_1, \ldots, \alpha_i) | c$.

How far we can push it?

Goldmann, Russell, *I&C 2002*

If $\mathbf{G}$ is a finite nilpotent group then $\textsc{PolSat}(\mathbf{G}) \in \mathrm{P}$.
If $\mathbf{G}$ is a finite non-solvable group then $\textsc{PolSat}(\mathbf{G})$ is NP-complete.



Nilpotent, solvable

Nilpotent sequence

$$\begin{aligned} \mathbf{G}^{(0)} &= \mathbf{G} \\ \mathbf{G}^{(k+1)} &= [\mathbf{G}^{(k)}, \mathbf{G}] \end{aligned}$$

$\mathbf{G}$ is k-nilpotent whenever k is minimal such that $\mathbf{G}^{(k)} = \{1_{\mathbf{G}}\}$.

Solvable sequence

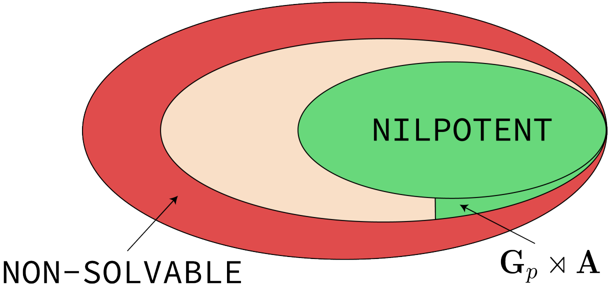
$$\begin{aligned} \mathbf{G}^{[0]} &= \mathbf{G} \\ \mathbf{G}^{[k+1]} &= [\mathbf{G}^{[k]}, \mathbf{G}^{[k]}] \end{aligned}$$

$\mathbf{G}$ is k-solvable whenever k is minimal such that $\mathbf{G}^{[k]} = \{1_{\mathbf{G}}\}$.

Grupy

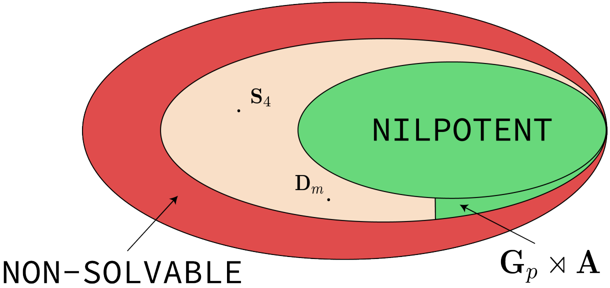
Horváth, Földvári, *IJAC 2019*

If $\mathbf{G}_p$ is a p-group and $\mathbf{A}$ is an abelian group, then any semidirect product $\mathbf{G}_p \rtimes \mathbf{A}$ has PolSat in P.

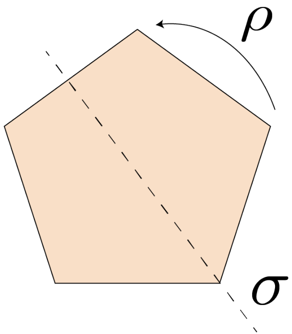


Horváth, Földvári, *IJAC 2019*

If $\mathbf{G}_p$ is a p-group and $\mathbf{A}$ is an abelian group, then any semidirect product $\mathbf{G}_p \rtimes \mathbf{A}$ has PolSat in P.



$\mathbf{D}_m$ - group of symmetries of regular m-gon



$2m$ symmetries:

- m rotations
- m reflections

$\mathbf{D}_m$ - group of symmetries of regular m-gon

Idziak, PK, Krzaczkowski, Weiß, *ICALP 2022*

$\textsc{PolSat}(\mathbf{D}_m)$ has a probabilistic polynomial-time algorithm iff m has at most one odd prime divisor assuming rETH.

Exponential Time Hypothesis (ETH)

Any algorithm solving 3-SAT requires time at least $2^{\Omega(n)}$.

Note: ETH is a stronger version of $P \neq NP$.

Note: current best algorithm for 3-SAT has complexity $O(1.321^n)$ (Hertli, Moser, Scheder 2011)

$\mathbf{D}_m$ - group of symmetries of regular m-gon

Idziak, PK, Krzaczkowski, Weiß, *ICALP 2022*

$\textsc{PolSat}(\mathbf{D}_m)$ has a probabilistic polynomial-time algorithm iff m has at most one odd prime divisor assuming rETH.

Normal strategy: to prove that problem B is NP-hard, show a **polynomial-time** reduction from 3-SAT to B.

Different strategy: to prove that problem B is not in P, show a **subexponential-time** reduction from 3-SAT to B.

Here: if r is the number of odd prime divisors of m, we do a $2^{O(n^{1/r}\log n)}$ time reduction from 3-SAT to $\textsc{PolSat}(\mathbf{D}_m)$. So we can't have $n^{o((\log n)^{r-1}/\log\log n)}$ algorithm for $\textsc{PolSat}(\mathbf{D}_m)$, or we would contradict ETH.

$\mathbf{D}_{2^\alpha p^\beta}$ - algorithm and generalizations

Note: the positive side $m = 2^\alpha \cdot p^\beta$ requires constructing a polynomial-time algorithm.

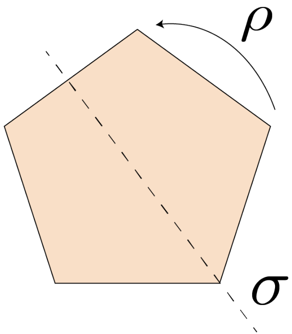
Algorithm: Assign random values to variables, check if the random assignment is a solution to the equation. Repeat poly(l) times, where l is the length of the equation.

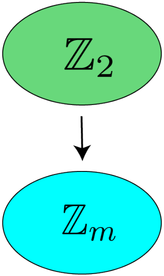
Idziak, PK, Krzaczkowski, Weiß, *ICALP 2022*

If a finite group $\mathbf{G}$ has a normal p-subgroup $\mathbf{G}_p$, such that $\mathbf{G}/\mathbf{G}_p$ is nilpotent then, assuming CDH, $\textsc{PolSat}(\mathbf{G})$ has a probabilistic polynomial-time algorihm.

Note: If $\mathbf{G}/\mathbf{G}_p$ is abelian we do not need CDH (Constant Degree Hyphothesis).

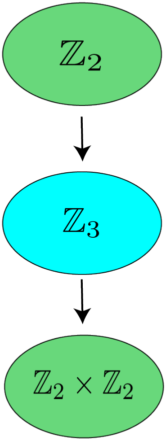
$\mathbf{D}_m$ - group of symmetries of regular m-gon





S_4 - group of permutations of 4-element set

$$S_4$$



$\mathbf{S}_4$ - group of permutations of 4-element set

Idziak, PK, Krzaczkowski, *LICS 2020*

$\textsc{PolSat}(\mathbf{S}_4)$ can not be solved faster than $n^{o(\log n)}$, assuming ETH.

Reason: the diagram for $\mathbf{S}_4$ has height 3.

How to formalize it? We say that the group $\mathbf{G}$ has a nipotent rank $\mathrm{nr}(\mathbf{G}) = h$, if h is the smallest number such that there is a sequence of normal subgroups of $\mathbf{G}$, such that
$\{1_{\mathbf{G}}\} = \mathbf{G}_0, \mathbf{G}_1, \ldots, \mathbf{G}_h = \mathbf{G}$ and each $\mathbf{G}_i/\mathbf{G}_{i-1}$ is nilpotent.

$\mathbf{S}_4$ - group of permutations of 4-element set

Idziak, PK, Krzaczkowski, *LICS 2020*

$\textsc{PolSat}(\mathbf{S}_4)$ can not be solved faster than $n^{o(\log n)}$, assuming ETH.

Reason: the diagram for $\mathbf{S}_4$ has height 3.

How to formalize it? We say that the group $\mathbf{G}$ has a nipotent rank $\mathrm{nr}(\mathbf{G}) = h$, if h is the smallest number such that there is a sequence of normal subgroups of $\mathbf{G}$, such that
$\{1_{\mathbf{G}}\} = \mathbf{G}_0, \mathbf{G}_1, \ldots, \mathbf{G}_h = \mathbf{G}$ and each $\mathbf{G}_i/\mathbf{G}_{i-1}$ is nilpotent.

Nilpotent rank

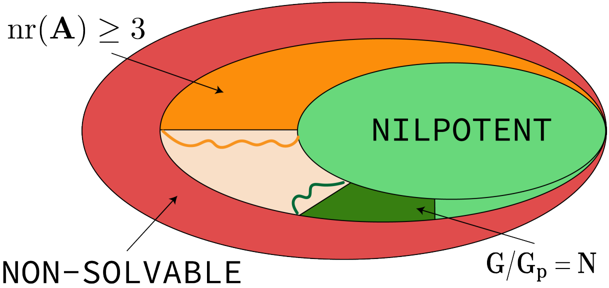
Idziak, PK, Krzaczkowski, Weiß, *TOCS 2022*

Assuming ETH, for a group $\mathbf{G}$ with a nilpotent rank $\mathrm{nr}(\mathbf{G}) \geqslant 3$ the problem $\textsc{PolSat}(\mathbf{G})$ has no polynomial-time solution.

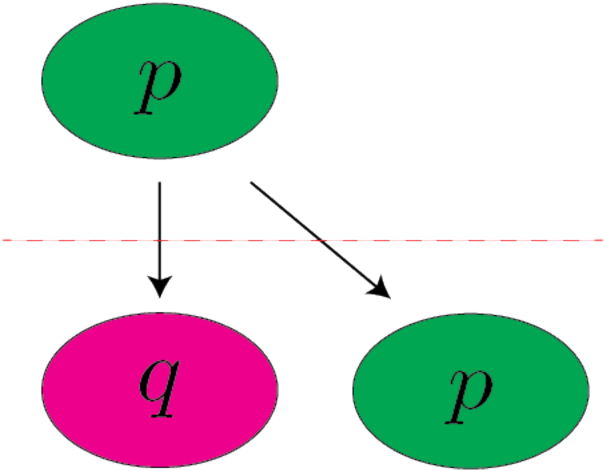
Note: it can be generalized to structures beyond groups.

So the only unresolved groups G, are the groups of nilpotent rank 2, i.e. they have nilpotent normal subgroup H such that G/H is also nilpotent.

So far so good



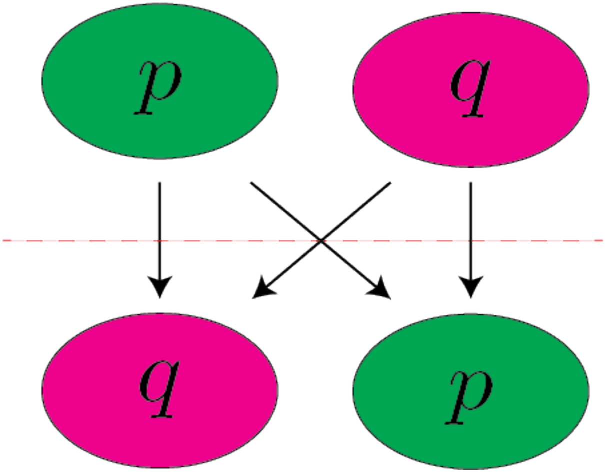
Looking for other poly-time groups



Example ($p = 2$):
$\mathbb{D}_{2^\alpha \boldsymbol{q}^\beta}$

Algorithm:
Take some random assignments.

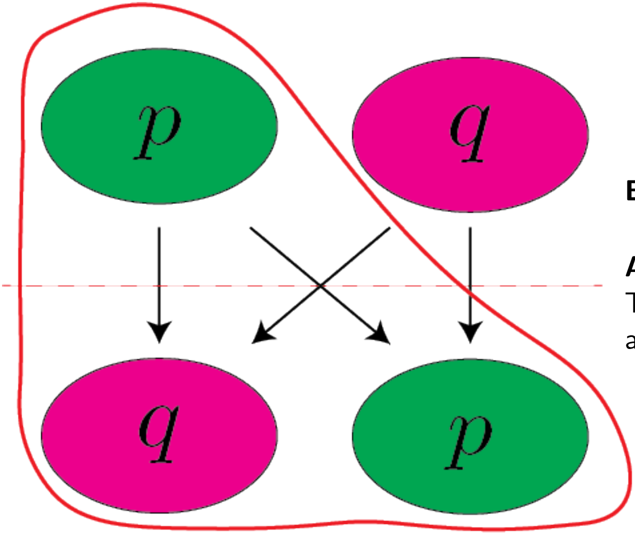
Looking for other poly-time groups



Example: $\mathbb{Z}_{pq} \wr \mathbb{Z}_{pq}$

Algorithm:
Take some random assignments.

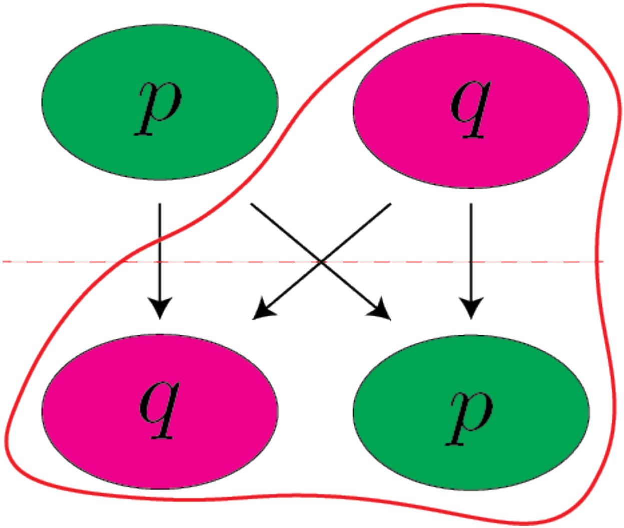
Looking for other poly-time groups



Example: $\mathbb{Z}_{pq} \wr \mathbb{Z}_{pq}$

Algorithm:
Take some random assignments.

Looking for other poly-time groups



Example: $\mathbb{Z}_{pq} \wr \mathbb{Z}_{pq}$

Algorithm:
Take some random assignments.

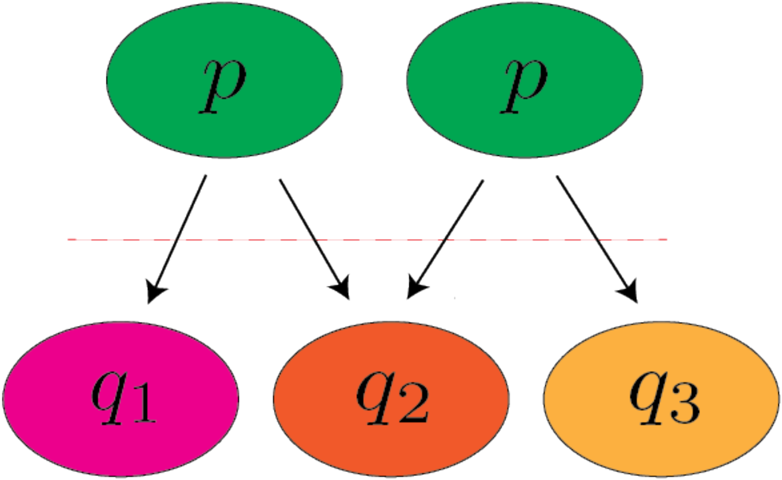
Two primes theorem

Idziak, PK, Krzaczkowski, Weiß, *ICALP 2022*

Let **G** be a finite group with $\mathrm{nr}(\mathbf{G}) = 2$ such that $|\mathbf{G}|$ has two prime divisors. Then $\textsc{PolSat}(\mathbf{G})$ is solvable in polynomial time.

It does not work with 3 primes anymore!

Messy example

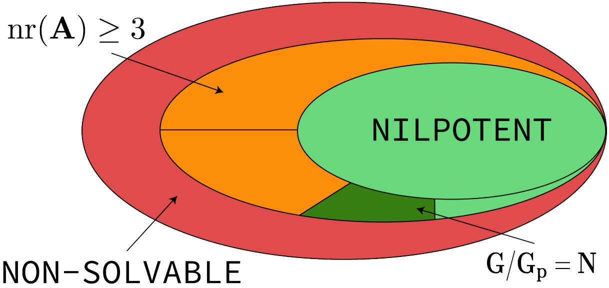


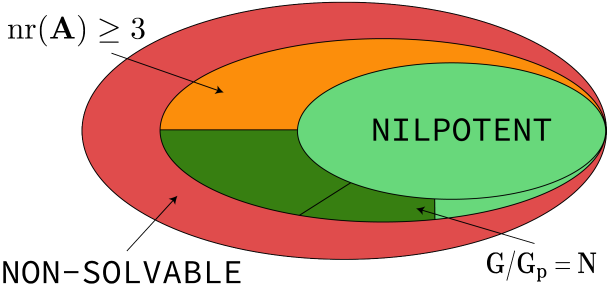
Some other connected problems

$\textsc{PolSat}(\mathbf{G})$ is problematic... But...

$\textsc{ListPolSat}(\mathbf{G})$ - we get an equation $\mathbf{p}(x_1, \ldots, x_n) = \mathbf{q}(x_1, \ldots, x_n)$, but we also have additional conditions on variables, i.e. for each variable x_i we get a list $L_i \subseteq G$ of allowed values, and we want a solution such that $x_i \in L_i$

$\textsc{PolEqv}(\mathbf{G})$ - we get an equation $\mathbf{p}(x_1, \ldots, x_n) = \mathbf{q}(x_1, \ldots, x_n)$ and we want to check that it is an identity, i.e. it is satisfied for all $(x_1, x_2, \ldots, x_n) \in G^n$


$\mathrm{nr}(\mathbf{A}) \geq 3$
NILPOTENT
NON-SOLVABLE
$\mathrm{G}/\mathrm{G_p} = \mathrm{N}$


$\mathrm{nr}(\mathbf{A}) \geq 3$
NILPOTENT
NON-SOLVABLE
$\mathrm{G}/\mathrm{G_p} = \mathrm{N}$

$\textsc{PolSat}$ - properties

Fakt (Horváth, Szabó, *JP&AA* 2012)

For alternating group $\mathbf{A}_4$:

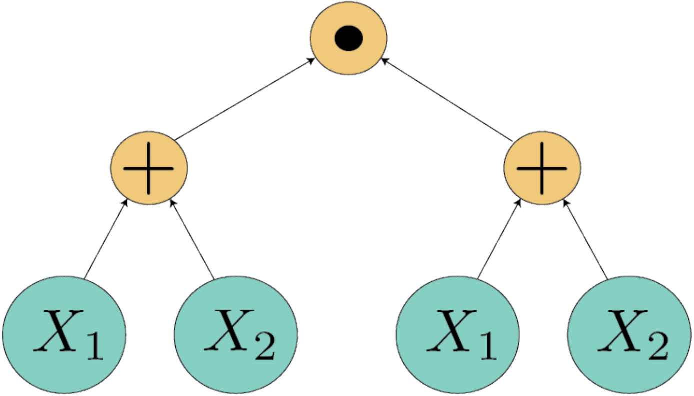
- $\textsc{PolSat}(\mathbf{A}_4; \cdot, ^{-1})$ is in P,
- $\textsc{PolSat}(\mathbf{A}_4; \cdot, ^{-1}, [x, y])$ is NP-complete.

Writing $[x_1, [x_2, [x_3, \ldots, [x_{n-1}, x_n] \ldots]$ with pute group operations requires exponential size in terms of n, but using commutator $[x, y]$ we can do it efficiently (as we can see).

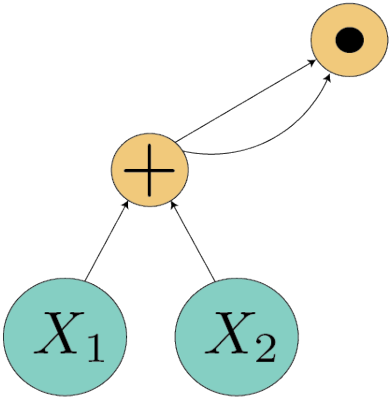
We cannot expect a general classification based on the algebraic properties of a structure!

Solution: use circuits instead of terms to represent polynomials.

$(x_1 + x_2) \cdot (x_1 + x_2)$ - term representation



$(x_1 + x_2) \cdot (x_1 + x_2)$ - circuit representation



$\textsc{CSat}$ - basics

Goldmann, Russell, *I&C 2002*; Horváth, Szabó, *DM&TCS 2011*

For a finite group $\mathbf{G}$ the problem $\textsc{CSat}(\mathbf{G}) \in P$ if $\mathbf{G}$ is nilpotent.
Otherwise $\textsc{CSat}(\mathbf{G})$ is NP-complete.

Goldmann, Russell, *I&C 2002*; Horváth, *Alg. Univ. 2011*

For a finite ring $\mathbf{R}$ the problem $\textsc{CSat}(\mathbf{R}) \in P$ if $\mathbf{R}$ is nilpotent.
Otherwise $\textsc{CSat}(\mathbf{R})$ is NP-complete.

Schwarz, *STACS 2004*

For a finite lattice $\mathbf{L}$ the problem $\textsc{CSat}(\mathbf{L}) \in P$ if $\mathbf{L}$ is distributive.
Otherwise $\textsc{CSat}(\mathbf{L})$ is NP-complete.

CSAT - some informations

Idziak, Krzaczkowski, *LICS 2018*

For a finite algebra $\mathbf{A}$ from a
Congruence Modular (CM) variety one of the two conditions holds.

- $\mathrm{CSAT}(\mathbf{A}/\alpha)$ is NP-complete, for some congruence α of $\mathbf{A}$.
- $\mathrm{CSAT}(\mathbf{A})$ decomposes into a direct product
  DL-like $\times$ nilpotent

Nilpotent algebras are far more complex than nilpotent groups. For instance they do not decompose into a product of algebras of prime power size.

Idziak, Krzaczkowski *LICS 2018*; Kompatscher, *IJAC 2018*

If a finite algebra $\mathbf{A}$ from CM is not only nilpotent,
but also supernilpotent, then $\mathrm{CSAT}(\mathbf{A}) \in \mathrm{P}$.

$\textsc{CSat}$ - supernilpotent rank

A measure of complexity of a group was nilpotent rank.

For general algebras the better measure is a supernilpotent rank.

Kompatscher, 2020

For a finite nilpotent algebra $\mathbf{A}$ from CM of supernilpotent rank $h \geqslant 3$ the problem $\textsc{CSat}(\mathbf{A})$
cannot be solved faster than $n^{o(\log^{h-2} n)}$.

$\textsc{CSat}$ - comparison to $\textsc{PolSat}$

Idziak, PK, Krzaczkowski, 2023

If a finite algebra $\mathbf{A}$ from CM has a supernilpotent congrunce α with classes of size p^α,
such that $\mathbf{A}/\alpha$ is supernilpotent then, assuming CDH,
$\textsc{CSat}(\mathbf{A})$ has a probabilistic polynomial-time algorihm.

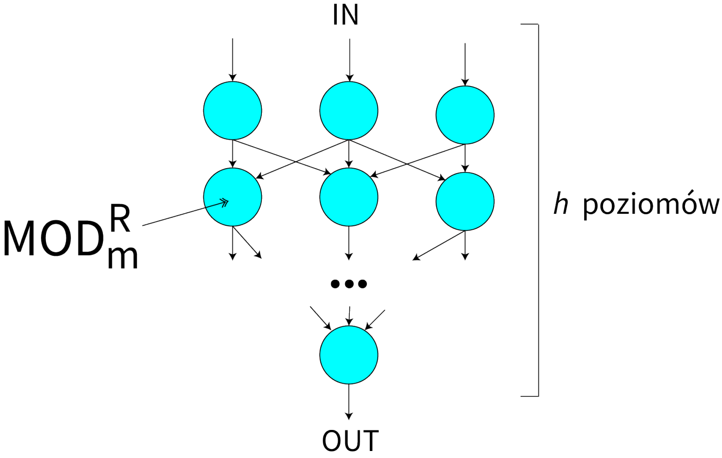
Idziak, PK, Krzaczkowski, Weiß, *ICALP 2022*

If a finite group $\mathbf{G}$ has a normal p-subgroup $\mathbf{G}_p$,
such that $\mathbf{G}/\mathbf{G}_p$ is nilpotent then, assuming CDH,
$\textsc{PolSat}(\mathbf{G})$ has a probabilistic polynomial-time algorihm.

That being said, for CSat there is no Two Primes Theorem. So although the combinatorics are similar, there are differences...

What is this CDH?

To understand CDH we first need to understand what is $CC_h[m]$ circuit. These circuits represent Boolean functions $\{0,1\}^n \longmapsto \{0,1\}$



$CC_h[m]$ - connections

Spooky sentence: $CC_h[m]$ circuits are quite good at simulating circuits over algebras of supernilpotent rank h. They are also quite good at simulating polynomials over groups of nilpotent rank h. Here m corresponds to the size of algebra/group.

Exponential Size Hyphothesis: for fixed h, m, $CC_h[m]$ circuits need size $\Omega(2^{n^c})$ to represent AND_n, for some constant c depending on h, m.

Fun fact: if the hyphothesis is true, then we get algorithms for $\textsc{PolSat}$ over solvable groups of quasipolynomial time complexity $2^{(\log n)^d}$. We also get similar algorithm for $\textsc{CSat}$ over nilpotent algebras.

CDH

Constant Degree Hypothesis: Any 3-level $\mathrm{MOD}_p \circ \mathrm{MOD}_m \circ \mathrm{AND}_d$ circuit requires size $2^{\Omega(n)}$ to compute AND_n.

Idziak, PK, Krzaczkowski, 2023

If a finite algebra **A** from CM has a supernilpotent congrunce α with classes of size p^α,
such that $\mathbf{A}/\alpha$ is supernilpotent then, assuming CDH,
$\textsc{CSat}(\mathbf{A})$ has a probabilistic polynomial-time algorihm.

Idziak, PK, Krzaczkowski, Weiß, *ICALP 2022*

If a finite group **G** has a normal p-subgroup $\mathbf{G}_p$,
such that $\mathbf{G}/\mathbf{G}_p$ is nilpotent then, assuming CDH,
$\textsc{PolSat}(\mathbf{G})$ has a probabilistic polynomial-time algorihm.

$E=mc^2$

$IP = PSPACE$

$x+y = y+x$

$F=ma$

Equations

$\Delta = b^2-4ac$

$e^{\iota\pi}+1 = 0$

$pV = nRt$

$\mathcal{PT}+\mathcal{DP} = ♥$

$\alpha+\beta+\gamma=180°$

$a^2+b^2=c^2$

Funding

Funding statement: Funded by the European Union (ERC, POCOCOP, 101071674). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.