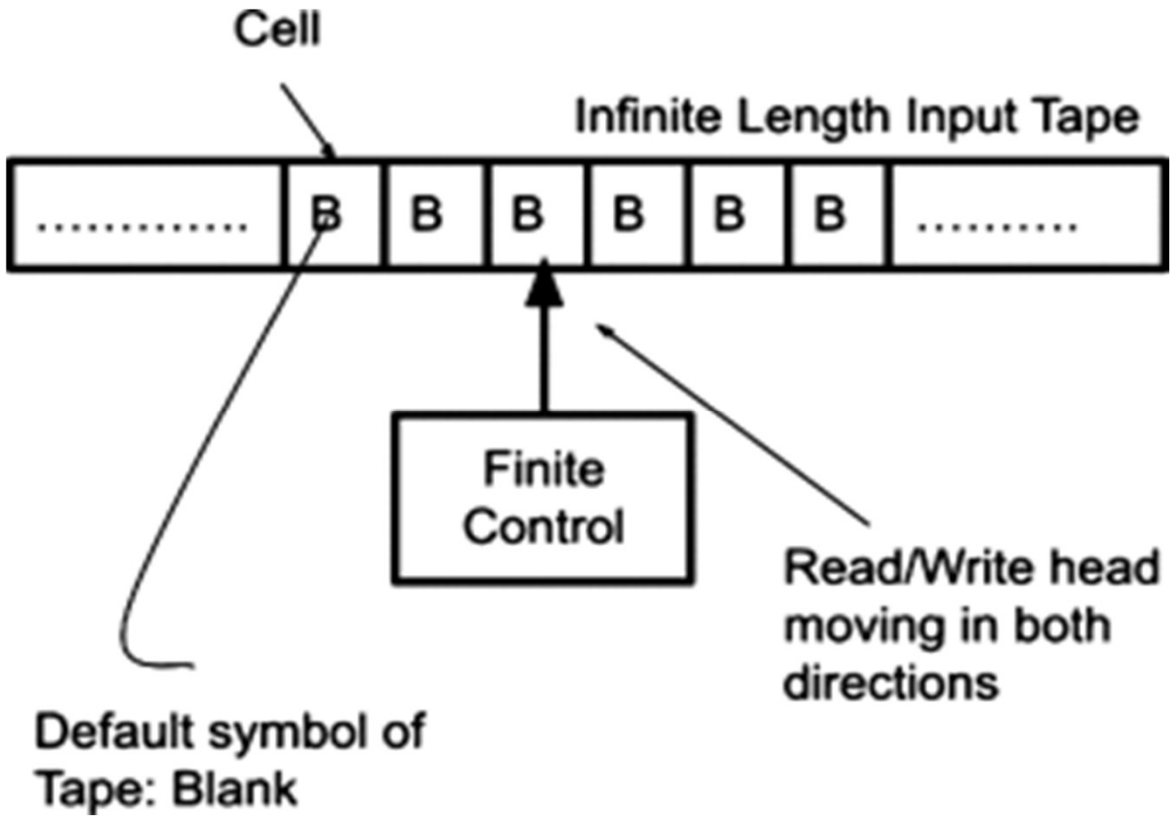
CIRCUIT COMPLEXITY CLASSES FROM FINITE ALGEBRAS

Piotr Kawałek

Tu Wien

TURING MACHINES



TURING MACHINES – AN ALTERNATIVE

Post's Theorem

Every function can be represented as a composition of AND, OR, ~ and projections (variables)

Corollary

For every language $L \subseteq \{0,1\}^*$ there is a sequence of Boolean expressions $(e_i)_{i \in \mathbb{N}}$ such that for word w in L of legth n we have $e_n(w) = 1$ iff w in L.

ASSYMPTOTIC GROWTH OF SIZE OF EXPRESSIONS

Let us say we have a computational problem (language), say CLIQUE:

1. There is a family of expressions $(e_i)_{i \in \mathbb{N}}$ solving the problem.
2. But the sizes of $(e_i)_{i \in \mathbb{N}}$ very likely grow like $2^{\Omega(n)}$
3. The smaller the size of circuits we need, the easier the problem is.

RELATION TO PTIME

1. Every problem in PTIME also has Boolean circuits of polynomial-size.

RELATION TO PTIME

1. Every problem in PTIME also has Boolean circuits of polynomial-size.
2. One can think of circuits as of expressions in which we can reuse subexpressions. (they are represented by directed acyclic graphs instead of trees)

RELATION TO PTIME

1. Every problem in PTIME also has Boolean circuits of polynomial-size.
2. One can think of circuits as of expressions in which we can reuse subexpressions. (they are represented by directed acyclic graphs instead of trees)
3. We write P/poly for the class of problems we can solve with polynomial size Boolen circuits.

RELATION TO PTIME

1. Every problem in PTIME also has Boolean circuits of polynomial-size.
2. One can think of circuits as of expressions in which we can reuse subexpressions. (they are represented by directed acyclic graphs instead of trees)
3. We write P/poly for the class of problems we can solve with polynomial size Boolen circuits.
4. The class is called nonuniform P, because we have different circuits for different sizes.

CONNECTION TO P VS NP

If some NP-complete problem (CLIQUE) requires superpolynomial size circuits then $P \neq NP$.

There are no strong lower bound results for standard Boolean Circuits.

ALGEBRAIC RELATIVIZATION

One can take a finite algebra **A** with a finite set of basic operations.

Then we can study which languages are recognized by polynomial size circuits over **A**.

MONOTONE CIRCUITS

Say our algebra **A** is just a two element lattice with join and meet.

Karchmer, Wigderson 92

UCON – undirected graph connectivity has polynomial size monotone circuits.

Razbarov 85; Alon, Boppana 87

CLIQUE requires exponential size monotone circuits.

Cavalar et. al. 2025

PERFECT-MATCHING requires exponential size monotone circuits.

GENERAL ALGEBRAS

We will stick to Boolean languages: $L \subseteq \{0,1\}^*$

How to make a circuit over an algebra **A** to recognize Boolean language?

On the output: interpret the result with accepting set $S \subseteq A$

On the input: interpret values 0,1 as elements of A

NONUNIFORM P OVER AN ALGEBRA

We say that a language $L \subseteq \{0,1\}^*$ is in the class $\mathsf{nuP}_{\mathbf{A}}$ iff there is

1. a function $\iota : \{0,1\} \longrightarrow A$
2. an accepting set $S \subseteq A$
3. a sequence $(\mathbf{p}_i)_{i \in \mathbb{N}}$ of circuits over **A** of size O(poly(i))

Such that for each $w = b_1 b_2 \dots b_n \in \{0,1\}^*$ we have

$$\mathbf{p}_n(\iota(b_1), \iota(b_2), \dots, \iota(b_n)) \in S \text{ iff } w \in L$$

COLLECTION OF FACTS

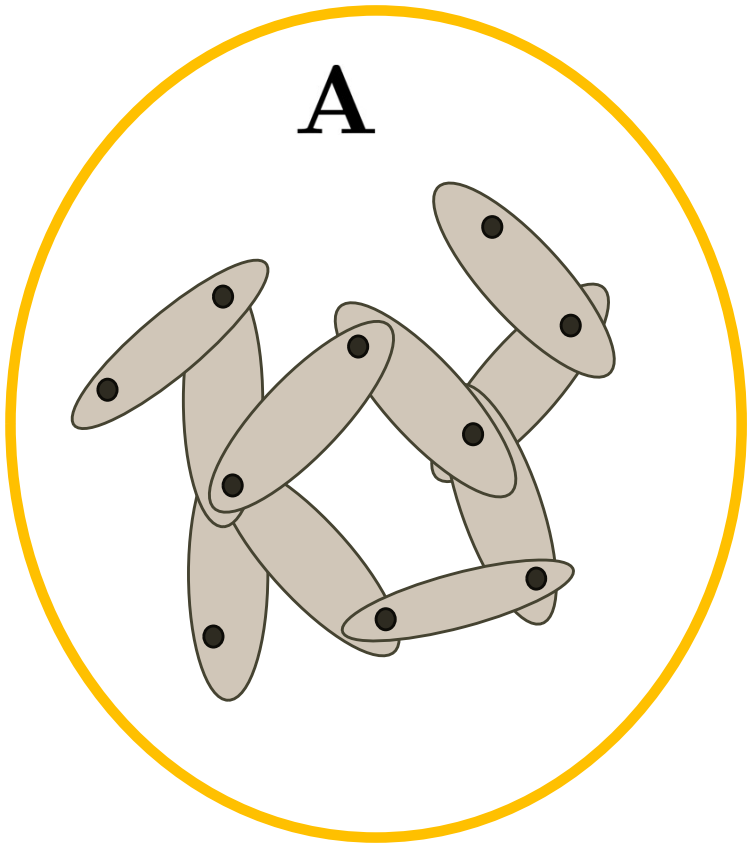
1. $\mathrm{nuP}_{\mathbf{A}}$ depends only on the polynomial clone of A (property of circuits and finitetely generated clones)
2. For a Boolean algebra **A** on two-element domain $\mathrm{nuP}_{\mathbf{A}} = \mathrm{P}/poly$, and it is the largest possible complexity class $\mathrm{nuP}_{\mathbf{A}}$ can reach.
3. For a two-element lattice **A** we have
$$\mathrm{nuP}_{\mathbf{A}} = \mathrm{MONOTONE} \cup \mathrm{ANTIMONOTONE}$$
4. $\mathrm{nuP}_{\mathbf{A}}$ can only be smaller when going to subalgebras and quotients, and it also works quite well with products.

COLLECTION OF FACTS

Some clones are too weak to even have complexity theory on them, for instance if **A** contains just projections and constants you cannot really solve any nontrivial problem with circuits over **A**

If **A** is from a congruence modular variety (CM) and is not supernilpotent it can compute at least all monotone functions (see Idziak, Kawałek, Krzaczkowski ICALP'25).

STRUCTURE OF SIMPLE ALGEBRAS



Minimal sets: minimal images of unary polynomial that are not one-element sets.

Combinatorics: minimal sets cover/connect A.

Representation: every $a \in A$ is uniquely determined by all projections $e(a)$ to minimal sets.

Algebra: Induced algebras on minimal sets are poly-equivalent.

5 TYPES OF SIMPLE ALGEBRAS

1. Unary, all polynomial operations of bounded arity,
2. Vector space (dimension 1), subreducts of affine structures,
3. Boolean Algebra, polynomially very reach,
4. Lattice, partially ordered,
5. Semilattice, partially ordered.

These 5 types generalize to covering pairs of congruences (prime quotients).

5 TYPES OF SIMPLE ALGEBRAS

1. Unary, $\mathrm{nuP}_{\mathbf{A}} = \mathrm{BAR}$
2. Vector space, $\mathrm{nuP}_{\mathbf{A}} = \mathrm{BAR} \circ \mathrm{MOD}_p$
3. Boolean Algebra, $\mathrm{nuP}_{\mathbf{A}} = \mathrm{P}/poly$
4. Lattice, if totally ordered $\mathrm{MONOTONE} \subseteq \mathrm{nuP}_{\mathbf{A}} \subseteq \mathrm{MULTIMONOTONE}$ otherwise $\mathrm{nuP}_{\mathbf{A}} = \mathrm{P}/poly$
5. Semilattice, complex behaviours: can encode MULTIMONOTONE , P/poly, fragments of AC^0, fragments of AC[p] and who knows what else .

WHAT TO DO FOR GENERAL ALGEBRAS

For CM we have only types 2,3,4 and the structure of minimal sets is clean.

Useful Fact

Let α be a congruence atom of A, then A is subreduct of

$$T^{[k]} \boxtimes \mathbf{A}/\alpha$$

Where T is a $(0, \alpha)$-trace (it is either one-dim vector space, two element Boolean Algebra or two element lattice).

CHARACTERIZATION FOR CM

Kawałek, Krzaczkowski 26

Let $\mathbf{A}$ be a finite algebra from a congruence modular variety, then
- If there is a two elemet subset U for which the clone of $\mathbf{A}_{|U}$ is full then $\mathrm{nuP}_{\mathbf{A}} = \mathrm{P}/poly$

CHARACTERIZATION FOR CM

Kawałek, Krzaczkowski 26

Let $\mathbf{A}$ be a finite algebra from a congruence modular variety, then

- Let all prime quotients of $\mathbf{A}$ be of type 4 (lattice). If $\mathbf{A}$ decomposes as a subdirect product of totally ordered algebras then $\mathrm{nuP}_{\mathbf{A}} \subseteq \mathrm{MULTIMONOTONE}$. Otherwise $\mathrm{nuP}_{\mathbf{A}} = \mathrm{P}/poly$.

CHARACTERIZATION FOR CM

Kawałek, Krzaczkowski 26

Let $\mathbf{A}$ be a finite algebra from a congruence modular variety, then

- If $\mathbf{A}$ is solvable Malcev algebra (type 2 only) $\mathrm{nuP}_{\mathbf{A}} \leq \mathrm{NC}^2$

CHARACTERIZATION FOR CM

Kawałek, Krzaczkowski 26

Let $\mathbf{A}$ be a finite algebra from a congruence modular variety, then

- If **A** has a congruence α such that $\mathbf{A}/\alpha$ decomposes into subdirect product of totally ordered algebras and all prime quotients below α are of type 2 (affine) then $\mathrm{nuP}_{\mathbf{A}} \subseteq \mathrm{NC}^2 \circ \mathrm{MONOTONE}$

CHARACTERIZATION FOR CM

Kawałek, Krzaczkowski 26

Let $\mathbf{A}$ be a finite algebra from a congruence modular variety, then

- Otherwise $\mathrm{nuP}_{\mathbf{A}} = \mathrm{P}/poly$

COMPLEXITY CLASSES AND CLASSES OF ALGEBRAS

It is sometimes better to consider complexity on the entire class/variety of algebras to get a precise description.

If $\mathcal{C}$ is the class of algebras then $\mathsf{nuP}_{\mathcal{C}} = \bigcup_{\mathbf{A} \in \mathcal{C}} \mathsf{nuP}_{\mathbf{A}}$

$\mathsf{nuP}_{\mathsf{SOLVABLE}} = \mathsf{nuItDet}$, $\mathsf{NC}^1 \leq \mathsf{nuItDet} \leq \mathsf{NC}^2$

$\mathsf{nuP}_{\mathsf{NILPOTENT}} = CC$

SOLVABLE ALGEBRAS AND DETERMINANTS

There exists a finite solvable Malcev algebra **A** of characteristic q with a congruence lattice of height 2 such that for any circuit **p** over **A** of size l with accepting set S and an interpretation ι the expression

$$\mathbf{p}(\iota(b_1), \ldots, \iota(b_n)) \in S$$

can be rewriten to a form det(**M**) =1, where **M** is a poly(l)-size matrix, filled with constant from GF(q) and variables b_i.

The oposite is also true and $\mathrm{nuP}_{\mathbf{A}} = \mathrm{nuDet}_q$.

DETERMINANTS AND SOLVABILITY

Given tight chain of congruences

$$0_{\mathbf{A}} = \alpha_0 \prec \alpha_1 \prec \ldots \prec \alpha_h = 1_{\mathbf{A}}$$

Such that $\alpha_{i-1} \prec \alpha_i$ is of characersistic p_i one can show that

$$\mathsf{nuP}_{\mathbf{A}} \subseteq \mathsf{nuDet}_{p_1} \circ \ldots \circ \mathsf{nuDet}_{p_h}$$

And hence by Berkowitz's algorithm $\mathsf{nuP}_{\mathbf{A}} \subseteq NC^2$

DETERMINANTS AND SOLVABILITY

Let

$$\mathsf{nuItDet} = \bigcup_{h \in \mathbb{N}} \bigcup_{p_1,..,p_h} \mathsf{nuDet}_{p_1} \circ .. \circ \mathsf{nuDet}_{p_h}$$

Then

$$\mathsf{nuP}_{\mathsf{SOLVABLE}} = \mathsf{nuItDet}$$

and

$$NC^1 \subseteq \mathsf{nuItDet} \subseteq NC^2$$

OTHER COMMENTS

1. If we use type 5 (semilattice type) we can express more natural classes studied in Computational Complexity, like AC, ACC, AC[p]. Type 5 is difficult to understand even for simple algebras.
2. One can study circuit complexity going beyond finite structures and finite signatures.
3. For a complexity class $\mathcal{C}$ with good properties one can define pseudoveriaty as all the algebras of a given signature that satisfy $\mathrm{nuP}_{\mathbf{A}} \subseteq \mathcal{C}$.

THANK YOU FOR FUNDING

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.