The Complexity of Intersecting Finite Automata Having Few Final States

Michael Blondin[1 2] Andreas Krebs[3] Pierre McKenzie[1]

[1]DIRO, Université de Montréal

[2]LSV, ENS Cachan

[3]Universität Tübingen

October 31, 2013

Definition

An *automaton* is a 5-tuple:

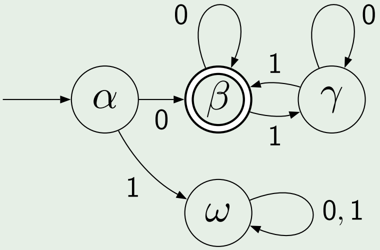
- Ω (finite set of *states*)
- Σ (finite *alphabet*)
- $\delta : \Omega \times \Sigma \to \Omega$ (*transition function*)
- $\alpha \in \Omega$ (*initial state*)
- $F \subseteq \Omega$ (*final states*)

Definition

Transition monoid $\mathcal{M}(A)$ of A:

$$\langle \{ T_\sigma \ : \ \sigma \in \Sigma \} \rangle \text{ where } T_\sigma(\gamma) = \delta(\gamma, \sigma).$$

Example



$$T_{011} = \begin{pmatrix} \alpha & \beta & \gamma & \omega \\ \beta & \beta & \gamma & \omega \end{pmatrix}$$

Definition

$\mathsf{AutoInt}_b(X)$ (Automata nonemptiness intersection problem)

Input: Automata $A_1, \ldots, A_k$ on alphabet Σ with $\mathcal{M}(A_i) \in X$ and at most b final states.

Question: $\bigcap_{i=1}^{k} \mathrm{Language}(A_i) \neq \emptyset$?

Definition

$\text{AutoInt}_b(\cup^m X)$ (Generalized automata intersection problem)

Input: Automata $A_{1,1}, \ldots, A_{k,m}$ on alphabet Σ with $\mathcal{M}(A_{i,j}) \in X$ and at most b final states.

Question: $\bigcap_{i=1}^{k} \bigcup_{j=1}^{m} \text{Language}(A_{i,j}) \neq \emptyset$?

Kozen 77

AutoInt and AutoInt_1 are PSPACE$-$complete.

Galil 76

AutoInt is NP$-$complete when $\Sigma = \{a\}$.

AutoInt interesting because generalizes:

Definition

Memb(X) (Membership problem)

Input: $g, g_1, \ldots, g_k : [m] \to [m]$ such that $\langle g_1, \ldots, g_k \rangle \in X$.
Question: $g \in \langle g_1, \ldots, g_k \rangle$?

AutoInt interesting because generalizes:

Definition

Memb(X) (Membership problem)

Input: $g, g_1, \ldots, g_k : [m] \to [m]$ such that $\langle g_1, \ldots, g_k \rangle \in X$.
Question: $g \in \langle g_1, \ldots, g_k \rangle$?

Connections with graph isomorphism led to deep results on group problems. It is known that Memb(Groups) $\in$ NC.

Definition

AC^k: languages accepted by Boolean circuits of poly size and depth $O(\log^k n)$. NC^k: similar with gates of indegree 2.

$$NC = AC = \bigcup_{k \geq 0} NC^k$$

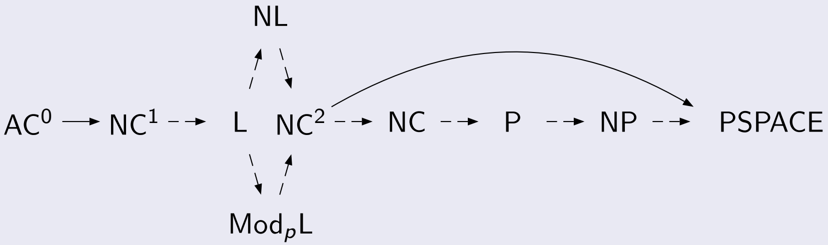
Definition

L: languages accepted by log-space Turing machines.

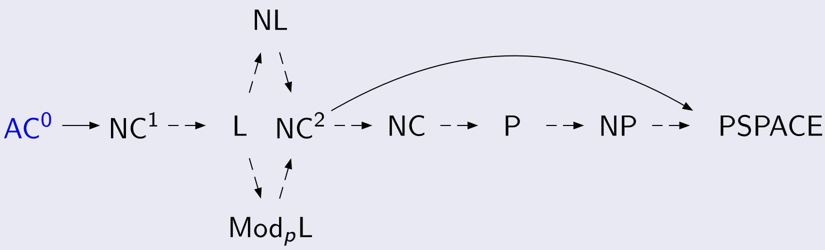
NL: languages accepted by log-space non deterministic Turing machines.

Mod_pL: languages S s.t. $w \in S$ iff $\#$ accept paths $\equiv 0 \pmod{p}$ for some NL machine.

Inclusion chain of complexity classes

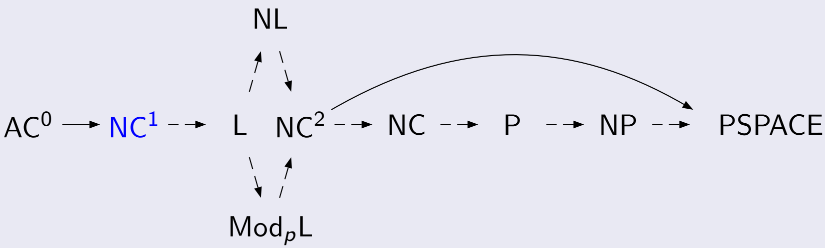


Inclusion chain of complexity classes



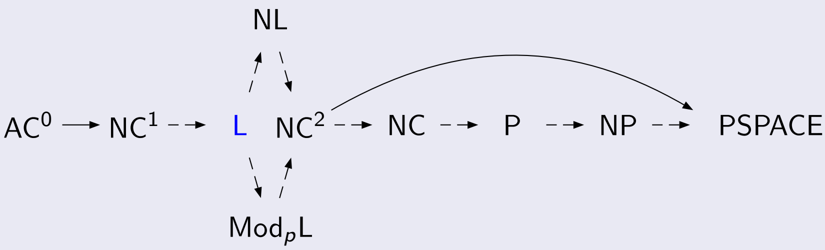
Contains: binary addition/substraction, star-free languages. Does not contain: parity/majority. Equals: FO(BIT), FO($+, \times$) where variables $=$ positions in words.

Inclusion chain of complexity classes



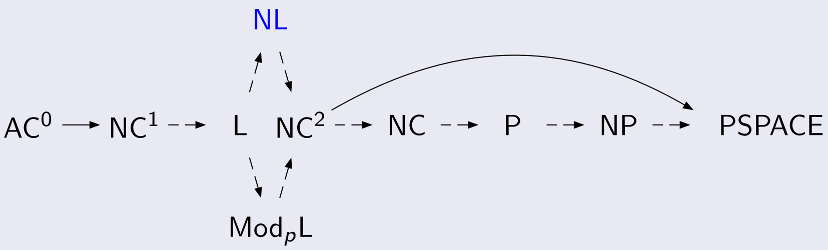
Contains: binary multiplication/division, regular languages, parity/majority.

Inclusion chain of complexity classes



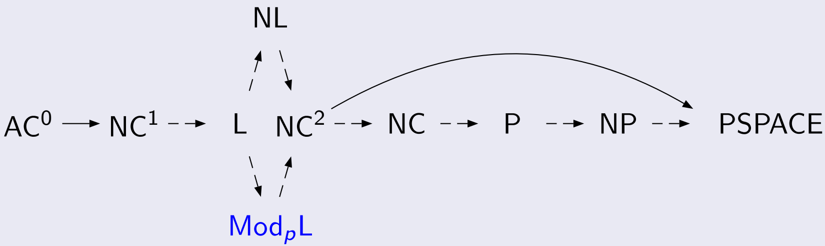
Complete problems: undirected connectivity, 2⊕SAT. Contains: problems defined in MSO on graphs of bounded tree-width.

Inclusion chain of complexity classes



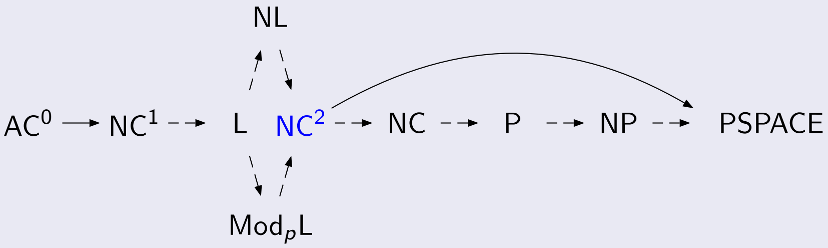
Complete problems: directed connectivity, 2SAT, testing an automaton for emptiness. Equals: coNL.

Inclusion chain of complexity classes



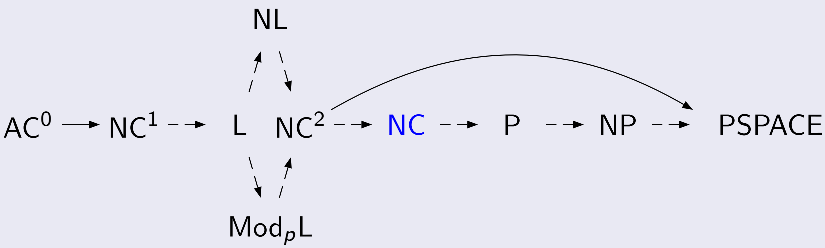
Complete problems: linear algebra mod p. Equals: coMod_pL.

Inclusion chain of complexity classes



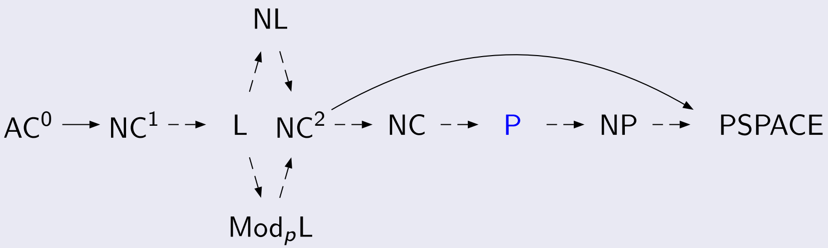
Contains: determinant, automata minimization.

Inclusion chain of complexity classes



Contains: membership in permutation groups.

Inclusion chain of complexity classes



Complete problems: circuit value problem, linear programming.

Main result: completeness results for $\mathrm{AutoInt}_b(X)$

	Maximum number of final states			
	1	2	1 with $\cup^2$	3+
$\Sigma = \{a\}$	L	L	NL	NP
$\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$	$\oplus$L	$\oplus$L	NP	NP
$\mathbb{Z}_p \times \cdots \times \mathbb{Z}_p$	$\mathrm{Mod}_p\mathrm{L}$	NP	NP	NP
Abelian groups	$\in \mathrm{NC}^3$	NP	NP	NP
Groups	$\in$ NC	NP	NP	NP
$\mathbf{J_1}$	$\in \mathrm{AC}^0$	NP	NP	NP

(white) Our classification.

(orange) Beaudry 88.

Complexity of $\text{AutoInt}_2(X)$

	Maximum number of final states			
	1	2	1 with $\cup^2$	3+
$\Sigma = \{a\}$	L	L	NL	NP
$\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$	$\oplus$L	$\oplus$L	NP	NP
$\mathbb{Z}_p \times \cdots \times \mathbb{Z}_p$	Mod_pL	NP	NP	NP
Abelian groups	$\in \text{NC}^3$	NP	NP	NP
Groups	$\in$ NC	NP	NP	NP
$\mathbf{J_1}$	$\in \text{AC}^0$	NP	NP	NP

Theorem

$\text{AutoInt}_2(X)$ *is hard for* NP *for any* X *beyond* $\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$.

Proof sketch

$X \not\subseteq \mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$ implies aperiodic monoid or cyclic group $\mathbb{Z}_q$, $q > 2$, in X.

Reduction from CIRCUIT–SAT to $\text{AutoInt}_2(X)$ in both cases.

Theorem

$\mathrm{AutoInt}_2(X)$ *is hard for* NP *for any* X *beyond* $\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$.

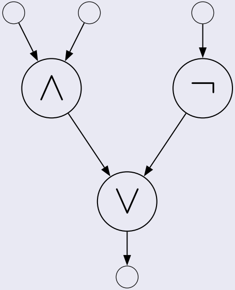
Proof sketch

$X \not\subseteq \mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$ implies aperiodic monoid or cyclic group $\mathbb{Z}_q$, $q > 2$, in X.

Reduction from CIRCUIT–SAT to $\mathrm{AutoInt}_2(X)$ in both cases.

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

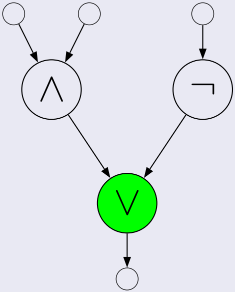
Given a circuit, we let Σ be the set of gates.



$\Sigma = \{\circ_0, \circ_1, \circ_2, \wedge_0, \neg_0, \vee_0, \circ_3\}$

Proof sketch: CIRCUIT–SAT reduces to $\mathrm{AutoInt}_2(\mathbb{Z}_q)$

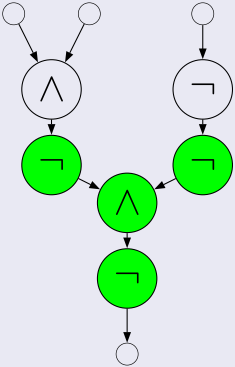
Given a circuit, we let Σ be the set of gates.



$\Sigma = \{\circ_0, \circ_1, \circ_2, \wedge_0, \neg_0, \vee_0, \circ_3\}$

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

Given a circuit, we let Σ be the set of gates.



$\Sigma = \{\circ_0, \circ_1, \circ_2, \wedge_0, \neg_0, \neg_1, \neg_2, \wedge_1, \circ_3\}$

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

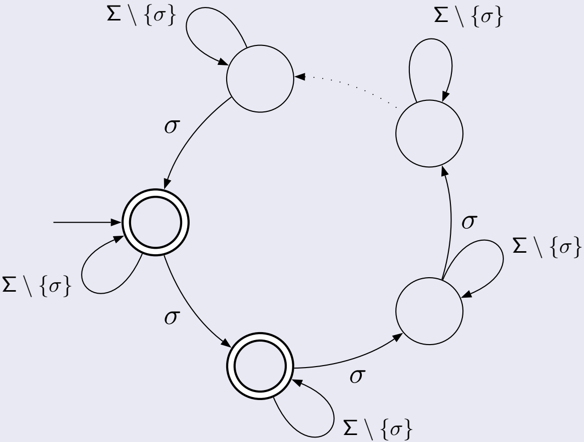
For each gate σ, we build automata A such that $\mathcal{M}(A) = \mathbb{Z}_q$.

Strategy:

- Occurrences of σ mod q encode assignment to σ (0 or 1),
- Automata verify soundness locally,
- Intersection represents satisfying assignments.

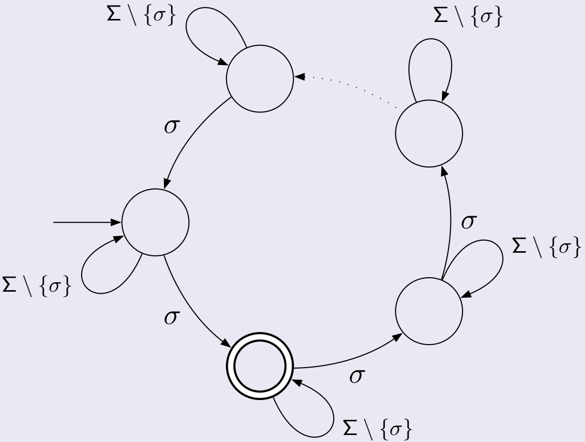
Proof sketch: CIRCUIT–SAT reduces to $\mathrm{AutoInt}_2(\mathbb{Z}_q)$

For each $\sigma \in \Sigma$, we accept words w such that $|w|_\sigma \equiv 0, 1 \pmod q$.



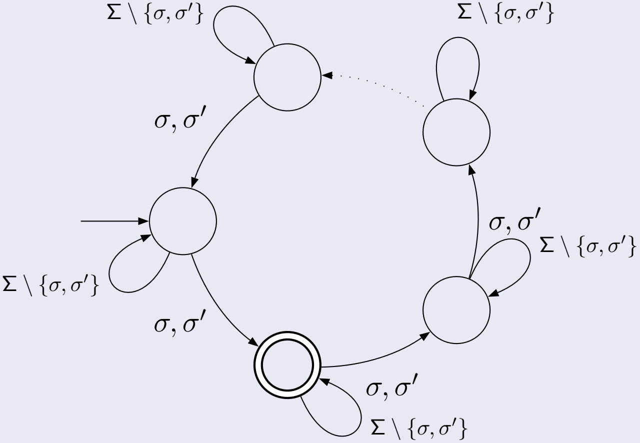
Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

For output gate σ, we accept words w such that $|w|_\sigma \equiv 1 \pmod{q}$.



Proof sketch: CIRCUIT–SAT reduces to AutoInt$_2(\mathbb{Z}_q)$

For each ¬-gate σ with input σ', we accept words w such that $|w|_\sigma + |w|_{\sigma'} \equiv 1 \pmod{q}$.



Proof sketch: CIRCUIT–SAT reduces to $\mathrm{AutoInt}_2(\mathbb{Z}_q)$

For each $\wedge$-gate σ with inputs σ', σ'', we accept words w such that $|w|_{\sigma'} + |w|_{\sigma''} - 2\,|w|_{\sigma} \equiv 0, 1 \pmod{q}$.

$\sigma'\sigma''\sigma$	$\sigma' \wedge \sigma''$	$\sigma' + \sigma'' - 2\sigma$
000	1	0
001	0	-2
010	1	1
011	0	-1
100	1	1
101	0	-1
110	0	2
111	1	0

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

For each $\wedge$-gate σ with inputs σ', σ'', we accept words w such that $|w|_{\sigma'} + |w|_{\sigma''} - 2\,|w|_{\sigma} \equiv 0, 1 \pmod q$.

$\sigma'\sigma''\sigma$	$\sigma' \wedge \sigma'' = \sigma$	$\sigma' + \sigma'' - 2\sigma \equiv 0, 1$
000	✓	✓
001	✗	✗
010	✓	✓
011	✗	✗
100	✓	✓
101	✗	✗
110	✗	✗
111	✓	✓

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

Problem when $q = 3$ since $-2 \equiv 1 \pmod 3$.

$\sigma'\sigma''\sigma$	$\sigma' \wedge \sigma'' = \sigma$	$\sigma' + \sigma'' - 2\sigma \equiv 0, 1$
000	✓	✓
001	0	-2
010	✓	✓
011	✗	✗
100	✓	✓
101	✗	✗
110	✗	✗
111	✓	✓

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

Problem when $q = 3$ since $-2 \equiv 1 \pmod 3$.

$\sigma'\sigma''\sigma$	$\sigma' \wedge \sigma'' = \sigma$	$\sigma' + \sigma'' - 2\sigma \equiv 0, 1$
000	✓	✓
001	✗	✓
010	✓	✓
011	✗	✗
100	✓	✓
101	✗	✗
110	✗	✗
111	✓	✓

Proof sketch: CIRCUIT–SAT reduces to $\mathsf{AutoInt}_2(\mathbb{Z}_q)$

When $q = 3$, we also build $|w|_{\sigma'} + |w|_{\sigma''} - |w|_{\sigma} \equiv 0, 1 \pmod 3$.

$\sigma'\sigma''\sigma$	$\sigma' \wedge \sigma''$	$\sigma' + \sigma'' - 2\sigma$	$\sigma' + \sigma'' - \sigma$
000	1	0	0
001	0	1	2
010	1	0	1
011	0	2	0
100	1	1	1
101	0	2	0
110	0	2	2
111	1	0	1

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

When $q = 3$, we also build $|w|_{\sigma'} + |w|_{\sigma''} - |w|_{\sigma} \equiv 0, 1 \pmod 3$.

$\sigma'\sigma''\sigma$	$\sigma' \wedge \sigma'' = \sigma$	$\sigma' + \sigma'' - 2\sigma \equiv 0, 1$	$\sigma' + \sigma'' - \sigma \equiv 0, 1$
000	✓	✓	✓
001	✗	✓	✗
010	✓	✓	✓
011	✗	✗	✓
100	✓	✓	✓
101	✗	✗	✓
110	✗	✗	✗
111	✓	✓	✓

Proof sketch: CIRCUIT–SAT reduces to $\text{AutoInt}_2(\mathbb{Z}_q)$

$\Rightarrow$) A satisfying assignment yields a word $\sigma_1^{b_1} \cdots \sigma_s^{b_s}$ accepted by the automata.

$\Leftarrow$) A word w accepted by the intersection yields a sastisfying assignment $\sigma_i \leftarrow |w|_{\sigma_i} \bmod q$. $\square$

Complexity of $\text{AutoInt}_1(\text{Abelian groups})$

	Maximum number of final states			
	1	2	1 with $\cup^2$	3+
$\Sigma = \{a\}$	L	L	NL	NP
$\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$	$\oplus$L	$\oplus$L	NP	NP
$\mathbb{Z}_p \times \cdots \times \mathbb{Z}_p$	Mod_pL	NP	NP	NP
Abelian groups	$\in NC^3$	NP	NP	NP
Groups	$\in$ NC	NP	NP	NP
$\mathbf{J_1}$	$\in AC^0$	NP	NP	NP

Definition

Let $A = (\Omega, \{\sigma_1, \ldots, \sigma_s\}, \delta, \alpha, F)$ be an abelian group automaton.

We define Φ_A as:

$$\left\{ v \in \mathbb{Z}_q^s \;:\; \delta(\alpha, \sigma_1^{v_1} \cdots \sigma_s^{v_s}) = \alpha \right\}$$

where $q = \mathrm{lcm}(\mathrm{ord}(T_{\sigma_1}), \ldots, \mathrm{ord}(T_{\sigma_s}))$.

Definition

Let U_A be a matrix such that its rows are a generating set for Φ_A.
Let $U_A^\perp$ be a matrix such that $U_A^\perp U_A^T \equiv 0 \pmod q$.

Lemma

Let $x, y \in \mathbb{N}^s$ *and* $w = \sigma_1^{x_1} \cdots \sigma_s^{x_s}$ *and* $w' = \sigma_1^{y_1} \cdots \sigma_s^{y_s}$, *then*

$$U_A^\perp x \equiv U_A^\perp y \pmod q \;\Leftrightarrow\; T_w(\alpha) = T_{w'}(\alpha).$$

Theorem

$\mathrm{AutoInt}_1(\textit{Abelian groups}) \in \mathrm{NC}^3$.

Proof sketch.

Let $A_1, \ldots, A_k$ be the given automata. We

- Compute U_{A_i} and $U_{A_i}^{\perp}$

Theorem

$\mathrm{AutoInt}_1(\textit{Abelian groups}) \in \mathrm{NC}^3$.

Proof sketch.

Let $A_1, \ldots, A_k$ be the given automata. We

- Compute U_{A_i} and $U_{A_i}^{\perp}$
- Compute $w_i \in \Sigma^*$ such that $T_{w_i}(\alpha_i) = \beta_i$

Theorem

$\mathrm{AutoInt}_1(\textit{Abelian groups}) \in \mathrm{NC}^3$.

Proof sketch.

Let $A_1, \ldots, A_k$ be the given automata. We

- Compute U_{A_i} and $U_{A_i}^{\perp}$
- Compute $w_i \in \Sigma^*$ such that $T_{w_i}(\alpha_i) = \beta_i$
- Verify $\exists x, \forall i \in [k]$, such that $U_{A_i}^{\perp} x \equiv U_{A_i}^{\perp} \begin{pmatrix} |w_i|_{\sigma_1} \\ \vdots \\ |w_i|_{\sigma_s} \end{pmatrix} \pmod{q_i}$.

Complexity of $\text{AutoInt}_2(\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2)$

	Maximum number of final states			
	1	2	1 with $\cup^2$	3+
$\Sigma = \{a\}$	L	L	NL	NP
$\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$	$\oplus$L	$\oplus$L	NP	NP
$\mathbb{Z}_p \times \cdots \times \mathbb{Z}_p$	Mod_pL	NP	NP	NP
Abelian groups	$\in \text{NC}^3$	NP	NP	NP
Groups	$\in$ NC	NP	NP	NP
$\mathbf{J_1}$	$\in \text{AC}^0$	NP	NP	NP

Hint for $\mathrm{AutoInt}_2(\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2) \in \oplus\mathrm{L}$

We can remove $\vee$ of such a system:

$$Bx \equiv b \pmod 2 \;\vee\; Bx \equiv b' \pmod 2$$

by introducing two variables

$$\begin{pmatrix} 0 & \cdots & 0 & 1 & 1 \\ B_{1,1} & \cdots & B_{1,s} & b_1 & b'_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ B_{m,1} & \cdots & B_{m,s} & b_m & b'_m \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \\ y \\ y' \end{pmatrix} \equiv \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \pmod 2$$

Gap from $\text{AutoInt}_2(\mathbb{Z}_2)$ to $\text{AutoInt}_2(\mathbb{Z}_q)$

	Maximum number of final states			
	1	2	1 with $\cup^2$	3+
$\Sigma = \{a\}$	L	L	NL	NP
$\mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$	$\oplus$L	$\oplus$L	NP	NP
$\mathbb{Z}_p \times \cdots \times \mathbb{Z}_p$	Mod_pL	NP	NP	NP
Abelian groups	$\in \text{NC}^3$	NP	NP	NP
Groups	$\in$ NC	NP	NP	NP
$\mathbf{J_1}$	$\in \text{AC}^0$	NP	NP	NP

- Relationships between algebraic problems and $\text{AutoInt}_b(X)$
- Extensive classification of AutoInt_b
- Close relationship between complexity of Memb and AutoInt_1
- Surprising gap from $\text{AutoInt}_2(\mathbb{Z}_2)$ to $\text{AutoInt}_2(\mathbb{Z}_3)$

What is the complexity of $\text{AutoInt}_1(X)$ for other X such that $\text{Memb}(X)$ is in between P and NP?

Thank you! Merci! Danke!