Logspace and compressed-word computations in nilpotent groups

Svetla Vassileva
Concordia University
Joint work with J. Macdonald, A. Miasnikov, A. Nikolaev

Group Theory International Webinar
February 5, 2015

The problems

For G finitely generated nilpotent group.

(I) Compute Mal'cev normal form.

(II) Membership problem.

(III) Compute the kernel of a homomorphism.

(IV) Compute subgroup presentations.

(V) Compute the centralizer of an element.

(VI) Conjugacy (search) problem.

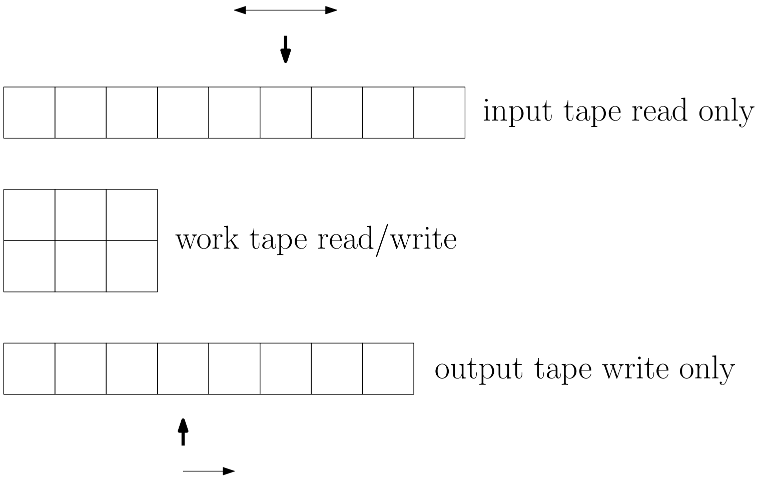
(Detailed descriptions to follow)

The results

1. Problems (I)-(VI) are decidable
   - in space $O(\log L)$
   - in time $O(L\log^3 L)$
2. We give polynomial bounds on the length of outputs.
3. Compressed-word versions of problems (I)-(VI) are decidable in polynomial time.

(Detailed theorems to follow)

Log-space transducers



Log-space $\Rightarrow$ P-time.

- Input length = n.
- Number of cells on work tape $\leq k \log n$.
- Configurations cannot be repeated.
- Total number of configurations $\sim 2^{k \log n} \sim n^k$
- Therefore, $O(n^k)$ time.
- P-time $\overset{?}{\Rightarrow}$ log-space: open problem.

Compressed words

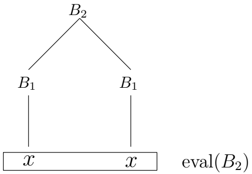
- Σ is a set of symbols, called *terminal symbols* with $\epsilon \in \Sigma$.
- A *straight-line program* or *compressed word* $\mathbb{A}$ over Σ consists of
  - $(\mathcal{A},<)$ – ordered finite set, called the set of *non-terminal symbols*,
  - exactly one *production rule* for each $A \in \mathcal{A}$ of the form
    - $A \to BC$ where $B, C \in \mathcal{A}$ and $B, C < A$ or
    - $A \to x$ where $x \in \Sigma$.
- The *root* is the greatest non-terminal.
- $\mathrm{eval}(\mathbb{A})$ is the word in Σ^* obtained by starting with the root non-terminal and successively replacing every non-terminal symbol with the right-hand side of its production rule.
- The *size*, $|\mathbb{A}|$, of $\mathbb{A}$ is the number of non-terminal symbols.

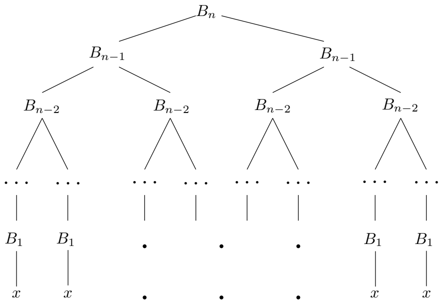
Example of compression

Consider the program $\mathbb{B}$ over $\{x\}$ with production rules

$$B_n \to B_{n-1}B_{n-1},\ B_{n-1} \to B_{n-2}B_{n-2},\ \ldots,\ B_1 \to x.$$

Unravel, $\text{eval}(B_2) = x^2$ and $\text{eval}(\mathbb{B}) = x^{2^{n-1}}$.





Nilpotent group

A group G is called *nilpotent* if it has a normal series

$$G = G_1 \rhd G_2 \rhd \ldots \rhd G_s \rhd G_{s+1} = 1 \tag{1}$$

such that

- $G_i/G_{i+1} \leq Z(G/G_{i+1})$ for all $i = 1, \ldots, s$, or, equivalently
- $[G, G_i] \leq G_{i+1}$ for all $i = 1, \ldots, s$.

Mal'cev basis

- G_i/G_{i+1} is abelian.
- For this talk, also torsion-free. However, results hold with torsion.
- $G_i/G_{i+1} = \langle a_{i1}, \ldots, a_{im_i} \rangle$ as an abelian group.
- $A = \{a_{11}, a_{12}, \ldots, a_{sm_s}\}$ is a polycyclic generating set for G
- Relabel A as $\{a_1, \ldots, a_m\}$.
- A is a *Mal'cev basis associated to the central series* (1).

Mal'cev normal forms

- Let $A = \{a_1, \ldots, a_m\}$ be a Mal'cev basis for G.
- Every element $g \in G$ may be written uniquely in the form

$$g = a_1^{\alpha_1} \ldots a_m^{\alpha_m},$$

where $\alpha_i \in \mathbb{Z}$.

- "Collect to the left" using relations $(i < j)$

$$a_j a_i = a_i a_j \cdot a_{j+1}^{\beta_{j+1}} \cdots a_m^{\beta_m}.$$

- $\mathrm{Coord}(g) = (\alpha_1, \ldots, \alpha_m)$ is the *coordinate tuple* of g.
- $a_1^{\alpha_1} \ldots a_m^{\alpha_m}$ is the *(Mal'cev) normal form* of g.
- Denote $\alpha_i = \mathrm{Coord}_i(g)$.

Working with Mal'cev coordinates

Let $\{a_1, \ldots, a_m\}$ be a Mal'cev basis for G. Then there are polynomials

$$p_1, \ldots, p_m, q_1, \ldots, q_m$$

such that for $\mathrm{Coord}(g) = (\gamma_1, \ldots, \gamma_m)$ and $\mathrm{Coord}(h) = (\delta_1, \ldots, \delta_m)$,

(i) $\mathrm{Coord}_i(gh) = p_i(\gamma_1, \ldots, \gamma_m, \delta_1, \ldots, \delta_m)$,

(ii) $\mathrm{Coord}_i(g^l) = q_i(\gamma_1, \ldots, \gamma_m, l)$, and

(iii) if $\mathrm{Coord}(g) = (0, \ldots, 0, \gamma_k, \ldots, \gamma_m)$, then

- *(a)* $\forall i < k$, $\mathrm{Coord}_i(gh) = \delta_i$ and $\mathrm{Coord}_k(gh) = \gamma_k + \delta_k$
- *(b)* $\forall i < k$, $\mathrm{Coord}_i(g^l) = 0$ and $\mathrm{Coord}_k(g^l) = l\gamma_k$.

  Example. $(a_1a_2a_3a_4a_5) \cdot (a_3^2a_4a_5) = a_1a_2a_3^3a_4^?a_5^?$.

Length bound for Mal'cev normal forms

Theorem

Let G be nilpotent group of class c with a Mal'cev basis A. Then, for any word w over A,

$$|\mathrm{Coord}_i(w)| \le \kappa |w|^c$$

where κ is a constant that depends only on the presentation of G.

- $|\mathrm{Coord}_i(w)|$ is the absolute value of the integer $\mathrm{Coord}_i(w)$;
- $|w|$ is the word length of w in terms of A.
- Number of bits of $\mathrm{Coord}(w)$ is $\sim \log |w|$ (so can store $\mathrm{Coord}(w)$ in memory).

Remark on nilpotent vs. polycyclic

Proposition

Let H be a polycyclic group with polycyclic generators $A = \{a_1, \ldots, a_m\}$. Suppose there is a polynomial $P(n)$ such that if w is a word over $A^{\pm 1}$ of length n then

$$|\mathrm{Coord}_i(w)| \leq P(n)$$

for all $i = 1, 2, \ldots, m$. Then H is virtually nilpotent.

Therefore, the results cannot be extended to polycyclic groups.

Usual vs. Mal'cev encoding

Consider $\mathbb{Z} = \langle a \rangle$.

- Encode a word w as $w = aaaaaaaaa$, so $|w| = 9$.
- Encode a word w as $w = a^9$, or, $w = 9$. So $\|w\| = \lceil \log_2 9 \rceil = 4$.

Similar with nilpotent groups. Let G have Mal'cev basis $a_1, \ldots, a_m$.

- Encode a word w as $w = a_{i_1} a_{i_2} \ldots a_{i_n}$. So $|w| = n$.
- This can be rewritten as $w = a_1 \ldots a_1 a_2 \ldots a_2 \cdots a_m \ldots a_m$.
- Here $|w| \sim n^c$.
- So $w = a_1^{\alpha_1} \ldots a_m^{\alpha_m}$ with $\alpha_1, \ldots \alpha_m \in \mathbb{Z}$.
- Encode $w = (\alpha_1, \ldots, \alpha_m) \in \mathbb{Z}^m$.
- Here $\|w\| \sim O(\log_2 n)$.

What about compressed words?

Working with compressed words

The strategy to do the compressed word version of problems is as follows.

- Convert the input SLPs to Mal'cev coordinates.
- Apply algorithms which work with Mal'cev coordinates in binary.
- Convert the output coordinate vectors to SLPs.

What about the size?

- Let L be the length of the SLP $\mathbb{A}$.
- The length of $\mathrm{eval}(\mathbb{A})$ is $\sim 2^L$.
- Each Mal'cev coordinate of $\mathrm{eval}(\mathbb{A})$ is $\sim 2^{cL}$.
- In binary, coordinates are $O(L)$ bits long.

Coordinate tuple $\longleftrightarrow$ SLP.

Theorem

Let G be a f.g. nilpotent group with Mal'cev generating set A.

- There is an algorithm that, given a straight-line program $\mathbb{A}$ over $A^{\pm}$, computes the coordinate vector $\mathrm{Coord}(\mathrm{eval}(\mathbb{A}))$.
- The algorithm runs in time $O(L^3)$, where $L = |\mathbb{A}|$.
- Each coordinate of $\mathrm{eval}(\mathbb{A})$ is expressed as a $O(L)$-bit number.

Compressed vs. usual input

- Input as words in generators.

$$\begin{array}{ccccc} w & \longrightarrow & a_1^{\alpha_1} \cdots a_m^{\alpha_m} & \xrightarrow{\text{binary}} & (\alpha_1, \ldots, \alpha_m) \\ L & \longrightarrow & L^c & \longrightarrow & \log L. \end{array}$$

- Input as compressed words.

$$\begin{array}{ccccccc} \mathbb{A} & \longrightarrow & \mathrm{eval}(\mathbb{A}) & \longrightarrow & a_1^{\alpha_1} \cdots a_m^{\alpha_m} & \xrightarrow{\text{binary}} & (\alpha_1, \ldots, \alpha_m) \\ L & \longrightarrow & 2^L & \longrightarrow & 2^{cL} & \longrightarrow & L. \end{array}$$

Computation of normal forms

Theorem

For every finitely generated nilpotent group G, the Mal'cev normal form of a word of length L is computable in

- space $O(\log(L))$ or
- time $O(L \cdot (\log L)^2)$

Proof

The algorithm – compute coordinates element by element.

- Denote $w = x_1 \cdots x_L$.
- Keep an array $\gamma = (\gamma_1, \ldots, \gamma_m)$ of coordinates in memory.
- At the end of step j, γ holds the coordinates of $x_1 \ldots x_j$.
- For $0 \leq j < L$, compute $\mathrm{Coord}(x_1 \cdots x_j x_{j+1})$ using the p_i with
  - $\mathrm{Coord}(x_1 \cdots x_j) = (\gamma_1, \ldots, \gamma_m)$ and
  - $\mathrm{Coord}(x_{j+1}) = (0, \ldots, 0, \pm 1, 0, \ldots, 0)$.

Complexity

- $|x_1 \cdots x_j| \leq L$, so $\gamma \leq \kappa L^c$ can be stored in logspace.
- $m(L-1)$ total evaluations of the polynomials p_i.
- Each evaluation of p_i requires arithmetic with $O(\log L)$-bit numbers, so can be performed in required space and time.

Compressed word problem

Corollary

The compressed word problem in every finitely generated nilpotent group is decidable in (sub)cubic time.

Note. Haubold, Lohrey, Mathissen had already observed that the compressed word problem is decidable in polynomial time. (Uses embedding in $UT_n(\mathbb{Z})$).

Matrix notation

Let G have Mal'cev basis $\{a_1, \ldots, a_m\}$.

$$\left\{\begin{array}{lcl} h_1 = a_1^{\alpha_{11}} & \cdots & a_m^{\alpha_{1m}} \\ \vdots & & \vdots \\ h_n = a_1^{\alpha_{1n}} & \cdots & a_m^{\alpha_{nm}} \end{array}\right. \leftrightsquigarrow \begin{pmatrix} \alpha_{11} & \cdots & \alpha_{1m} \\ \vdots & \ddots & \vdots \\ \alpha_{1n} & \cdots & \alpha_{nm} \end{pmatrix} = A.$$

- π_i is the column of the first non-zero entry ('pivot') in row i.
- $(h_1, \ldots, h_n)$ is in *standard form* if the matrix of coordinates A is in row-echelon form and entries above pivots are reduced.
- Denote $H = \langle h_1, \ldots, h_n \rangle$.
- $(h_1, \ldots, h_n)$ is *full* if for each $1 \le i \le m$, the subgroup $H \cap \langle a_i, a_{i+1}, \ldots, a_m \rangle$ is generated by $\{h_j \mid \pi_j \ge i\}$.

Uniqueness of standard form

Lemma [Sims]

Let $H \leq G$. There is a unique full sequence $U = (h_1, \ldots, h_s)$ that generates H. Further,

$$H = \{h_1^{\beta_1} \cdots h_s^{\beta_s} \mid \beta_i \in \mathbb{Z}\}$$

and $s \leq m$.

Goal: convert $(h_1, \ldots, h_n)$ to a sequence in standard form generating the same subgroup.

Uniqueness of standard form

Lemma [Sims]

Let $H \leq G$. There is a unique full sequence $U = (h_1, \ldots, h_s)$ that generates H. Further,

$$H = \{h_1^{\beta_1} \cdots h_s^{\beta_s} \mid \beta_i \in \mathbb{Z}\}$$

and $s \leq m$.

Goal: convert $(h_1, \ldots, h_n)$ to a sequence in standard form generating the same subgroup.

Matrix operations

Define three operations on tuples $(h_1, \ldots, h_n)$ of elements of G by their corresponding operations on the associated matrix are:

(1) swap row i with row j;

(2) replace row i by $\mathrm{Coord}(h_i h_j^N)$;

(3) add or remove a row of zeros.

All three of these operations preserve the subgroup $\langle h_1, \ldots, h_n \rangle$.

Row-reducing the matrix

Let A be an $n \times m$ matrix. Similar to row-reducing a matrix over $\mathbb{Z}$ (in fact, works same as over $\mathbb{Z}$ in the first column).

- Identify pivot.
- Use the gcd of the pivot column to clear out column.
- Number of operations $\sim n$.
- Repeat for each column (m times).
- Total number of operations $\sim mn$.

Magnitude of entries may increase

- Warning! When using the operation $h_i \to h_i h_j^N$, the magnitude of the largest entry may increase from M to M^d, $d =$ degree of multiplication polynomials.
- Greatest entry could be size $\sim M^{d^{mn}}$.

Length bound for reduced matrix

Lemma

Let $h_1, \ldots, h_n \in G$ and let R be the standard form of the associated matrix of coordinates. Then every entry, α_{ij}, of R is bounded by

$$|\alpha_{ij}| \leq CL^K,$$

where $L = |h_1| + \cdots + |h_n|$ is the total length of the given elements, and K and C are constants depending on G.

Proof relies on uniqueness of standard form.

Computing standard form

Lemma

There is an algorithm that, given $h_1, \ldots, h_n \in G$, computes the standard form of the matrix of coordinates in space logarithmic in $L = \sum_{i=1}^{n} |h_i|$ and in time $O(L \log^3 L)$.

- Start with $m \times m$ matrix (constant size).
- Reduce to standard form.
- Add a row and reduce (still constant size).
- Repeat until all n rows accounted for.
- Size never goes beyond $\sim 2m \times m$. Entries bounded.
- The size of the reduced matrix is $m \times m$.

Membership problem

Theorem

Let G be a finitely generated nilpotent group.
Let $h_1, \ldots, h_n \in G$ and $h \in G$.
Denote $L = |h| + |h_1| + \cdots + |h_n|$ and $H = \langle h_1, \ldots, h_n \rangle$.

- There is an algorithm that, decides whether or not $h \in H$.
- The algorithm runs in space $O(\log L)$ and time $O(L \log^3 L)$.
- If $h \in H$ the algorithm returns the unique expression $h = g_1^{\gamma_1} \cdots g_s^{\gamma_s}$, where $(g_1, \ldots, g_s)$ is the unique standard-form sequence for H, and the length of h is bounded by a degree $2m(6c^3)^m$ polynomial function of L.

Proof

- $(h_1, \ldots, h_n) \rightsquigarrow (g_1, \ldots, g_s)$.
- Here the g_i are in terms of the original Mal'cev basis.
- Denote $\mathrm{Coord}(h) = (\beta_1, \ldots, \beta_m)$.

- If $\beta_l \neq 0$ for some $1 \leq l < \pi_1$, then $h \notin H$.
- If $\mathrm{Coord}_{\pi_1}(g_1) \nmid \beta_{\pi_1}$, then $h \notin H$.
- Else, let

$$\gamma_1 = \frac{\beta_{\pi_1}}{\mathrm{Coord}_{\pi_1}(g_1)} \quad h' = g_1^{-\gamma_1} h.$$

- Repeat, replacing h by h' and $(g_1, \ldots, g_s)$ by $(g_2, \ldots, g_s)$.

Compressed word membership problem

Theorem

There is an algorithm that, given compressed words $\mathbb{A}_1, \ldots, \mathbb{A}_n, \mathbb{B}$ over a fixed finitely generated nilpotent group G, decides in time polynomial in $|\mathbb{B}| + |\mathbb{A}_1| + \ldots + |\mathbb{A}_n|$ whether or not $\mathrm{eval}(\mathbb{B})$ belongs to the subgroup generated by $\mathrm{eval}(\mathbb{A}_1), \ldots, \mathrm{eval}(\mathbb{A}_n)$.

Computing the kernel and pre-image of a homomorphism

- Let G and H be disjoint finitely generated nilpotent groups.
- Let $K = \langle g_1, \ldots, g_n \rangle \leq G$
- We specify a homomorphism $\phi : K \to H$ by a list of elements $h_1, \ldots, h_n \in H$ such that $\phi(g_i) = h_i$ for $i = 1, \ldots, n$.
- Denote $L = |h| + \sum_{i=1}^{m}(|h_i| + |g_i|)$.

Theorem

There is an algorithm that, given an element $h \in H$ guaranteed to be in the image of ϕ,

(i) computes a generating set X for the kernel of ϕ, and

(ii) computes an element $g \in G$ such that $\phi(g) = h$.

The algorithm runs in space $O(\log L)$ and time $O(L \log^3 L)$.

Computing subgroup presentation

Theorem

- Let G be a finitely presented nilpotent group.
- Let $g_1, \ldots, g_n$ be finite set of elements of G.
- Denote $L = \sum_{i=1}^{n} |g_i|$.

There is an algorithm that computes a presentation for the subgroup $\langle g_1, \ldots, g_n \rangle$. The algorithm runs in space $O(\log L)$ and time $O(L \log^3 L)$.

- Let $N = \langle x_1, \ldots, x_n \rangle$ be the free nilpotent group of class c.
- Define $\phi : N \to G$ by $x_i \mapsto g_i$.
- Compute $\ker \phi$.
- $N / \ker \phi \simeq \mathrm{im}\phi \simeq \langle g_1, \ldots, g_n \rangle$.

Presentation for compressed-word subgroups

Theorem

- Let G be a finitely presented nilpotent group.
- Let $\mathbb{A}_1, \ldots, \mathbb{A}_n$ be a finite set of straight-line programs over G.
- Denote $L = \sum_{i=1}^{n} |\mathbb{A}_i|$.

There is an algorithm that

- computes a presentation for $\langle \mathrm{eval}(\mathbb{A}_1), \ldots, \mathrm{eval}(\mathbb{A}_n) \rangle$,
- runs in time polynomial in L, and
- the size of the presentation is bounded by a polynomial of L.

Note. Size of presentation = number of generators plus sum of the lengths of the relators.

An example on encoding presentations for SLPs

- When working with SLPs, we get the relators as SLPs.
- How do we write down a presentation involving these relators?

Example. Suppose the following SLP is a relator.

$$\mathbb{A} = \{A_1 \to A_2A_3; \quad A_2 \to A_3A_4; \quad A_3 \to A_4A_4; \quad A_4 \to x\}.$$

Then $\mathrm{eval}(\mathbb{A}) = x^5$ and $|\mathrm{eval}(\mathbb{A})| \sim 2^L$.
To write a presentation using this relator we might do the following.

(1) $\langle x | xxxxx \rangle$ (but the length here is $\sim 2^L$), so bad. Or,

(2) $\langle x | \mathbb{A} \rangle$ (but this mixes encodings), so bad.

(3) $\left\langle x, a_1, a_2, a_3, a_4 | a_1 = 1, \begin{array}{ll} a_1 = a_2a_3, & a_2 = a_3a_4, \\ a_3 = a_4a_4, & a_4 = x \end{array} \right\rangle$. Size $O(L)$.

An example on encoding presentations for SLPs

- When working with SLPs, we get the relators as SLPs.
- How do we write down a presentation involving these relators?

Example. Suppose the following SLP is a relator.

$$\mathbb{A} = \{A_1 \to A_2A_3; \quad A_2 \to A_3A_4; \quad A_3 \to A_4A_4; \quad A_4 \to x\}.$$

Then $\mathrm{eval}(\mathbb{A}) = x^5$ and $|\mathrm{eval}(\mathbb{A})| \sim 2^L$.
To write a presentation using this relator we might do the following.

(1) $\langle x | xxxxx \rangle$ (but the length here is $\sim 2^L$), so bad. Or,

(2) $\langle x | \mathbb{A} \rangle$ (but this mixes encodings), so bad.

(3) $\left\langle x, a_1, a_2, a_3, a_4 | a_1 = 1, \begin{array}{ll} a_1 = a_2a_3, & a_2 = a_3a_4, \\ a_3 = a_4a_4, & a_4 = x \end{array} \right\rangle$. Size $O(L)$.

A note on the conjugacy problem in f.g. nilpotent groups

- A group is *conjugately separable* if whenever two elements are not conjugate, there is a finite quotient in which they are not conjugate.
- Gives rise to an enumerative algorithm to decide CP.
- F.g. nilpotent groups are conjugately separable (Remeslennikov ’69, Formanek ’76).
- Sims ’94 gave an algorithm based on matrix reductions and homomorphisms.
- Complexity not analysed.

Computing centralizers

Theorem

- Let G be a f.p. nilpotent group with Mal'cev basis of length m.
- Let $g \in G$.
- Denote $L = |g|$.

There is an algorithm that

- computes a generating set X for the centralizer of g in G,
- runs in space $O(\log L)$ and time $O(L \log^2 L)$.
- X contains at most m elements, and
- there is a degree $(6mc^2)^{m^2}$ polynomial function of L that bounds the length of each element of X.

The conjugacy problem is log-space decidable

Theorem

- Let G be a finitely presented nilpotent group.
- Let $g, h \in G$ be given as words.
- Denote $L = |g| + |h|$.

There is an algorithm that

- *(i)* produces $u \in G$ such that $g = u^{-1}hu$, or
  (ii) determines that no such element u exists,
- runs in space $O(\log L)$ and time $O(L \log^2 L)$, and
- the word length of u is bounded by a degree $2^m(6mc^2)^{m^2}$ polynomial function of L.

Compressed-word CP is polynomial-time decidable

Theorem

Let G be a finitely presented nilpotent group. There is an algorithm that, given two straight-line programs $\mathbb{A}$ and $\mathbb{B}$ over G, determines in time polynomial in $n = |\mathbb{A}| + |\mathbb{B}|$ whether or not $\mathrm{eval}(\mathbb{A})$ and $\mathrm{eval}(\mathbb{B})$ are conjugate in G. If so, a straight-line program over G of size polynomial in n producing a conjugating element is returned.