Circuit Complexity of Group Theoretic Problems

Kumulative Habilitationsschrift

vorgelegt von

Armin Weiß

Tag der Einreichung:

21. September 2020

Tag der mündlichen Habilitationsleistung:

21. April 2021

Contents

Zusammenfassung **V**

1 Introduction **1**

2 Preliminaries **5**

2.1 Complexity . . . 6

2.2 Algorithmic problems in group theory . . . 7

3 A logspace solution to the word and conjugacy problem of generalized Baumslag-Solitar groups **9**

3.1 Background . . . 9

3.2 Graphs of groups and GBS groups . . . 9

3.3 Results . . . 11

4 The conjugacy problem in free solvable groups and wreath products of abelian groups is in TC^0 **13**

4.1 Background . . . 13

4.2 Results . . . 13

4.3 Contribution by the author . . . 16

5 TC^0 circuits for algorithmic problems in nilpotent groups **17**

5.1 Background . . . 17

5.2 Preliminaries on nilpotent groups . . . 18

5.3 Results . . . 20

5.4 Contribution by the author . . . 22

6 The power word problem **23**

6.1 Background . . . 23

6.2 Results . . . 23

6.3 Outline of the proof of Theorem 6.2 . . . 25

6.4 Contribution by the author . . . 26

7 Groups with ALOGTIME-hard word problems and PSPACE-complete compressed word problems. **27**

7.1 Background . . . 27

7.2 Results . . . 28

7.3 Contribution by the author . . . 31

8 Hardness of equations over finite solvable groups under the exponential time hypothesis. **33**

8.1 Background . . . 33

8.2 Results . . . 34

8.3 Proof outline 36

List of publications summarized in this Habilitationsschrift **39**

Bibliography **41**

Zusammenfassung

Eines der grundlegendsten Probleme der algorithmischen Gruppentheorie ist das Wortproblem, das erstmals 1911 von Dehn beschrieben wurde [Deh11]. Für eine feste endlich erzeugte Gruppe[1] G ist das Wortproblem $\mathrm{WP}(G)$ die folgende Fragestellung: Gegeben ein Wort w über den Erzeugern von G, stellt w das neutrale Element von G dar? Obwohl das Wortproblem im Allgemeinen unentscheidbar ist [Nov55, Boo59], gibt es viele Klassen von Gruppen mit (effizient) entscheidbaren Wortproblemen, wie Dehn selbst auch feststellte. Ein bekanntes Beispiel hierfür sind lineare Gruppen, deren Wortprobleme deterministisch in logarithmischem Platz ($\mathsf{LOGSPACE}$) lösbar sind, wie Lipton und Zalcstein [LZ77] sowie Simon [Sim79] zeigten.

Tatsächlich gibt es auch einige Resultate über Gruppen mit Wortproblemen in noch kleineren Komplexitätsklassen als $\mathsf{LOGSPACE}$. Insbesondere betrifft dies endliche Gruppen: Das Wortproblem jeder endlichen, nicht-auflösbaren Gruppe ist NC^1-vollständig[2] [Bar89]; dagegen ist das Wortproblem jeder endlichen, auflösbaren Gruppe schon in ACC^0 [BT88][3]. Ferner ist das Wortproblem endlicher p-Gruppen in $\mathsf{ACC}^0[p]$ – und daher nach Smolenskys unterer Schranke [Smo87] beweisbar einfacher als das Wortproblem endlicher nicht-auflösbarer Gruppen.

Es gibt aber auch zahlreiche unendliche Gruppen mit ähnlich einfachen Wortproblemen: Robinson [Rob93] zeigte, dass das Wortproblem jeder nilpotenten Gruppe in TC^0 ist[4]. König und Lohrey verallgemeinerten dieses Ergebnis, indem sie zeigten, dass das Wortproblem jeder auflösbaren linearen Gruppe in TC^0 ist. Ferner ist die Klasse der endlich erzeugten Gruppen mit Wortproblemen in TC^0 abgeschlossen unter Kranzprodukten: Waack [Waa90] beschrieb hierfür zunächst eine NC^1-Reduktion, in [3][5] (siehe Kapitel 4 dieser Zusammenfassung) wird schließlich explizit eine TC^0-Reduktion bewiesen.

Andererseits ist keine nicht-auflösbare Gruppe bekannt, deren Wortproblem in TC^0 ist. Darüber hinaus ist das Wortproblem von (nicht-abelschen) freien Gruppen NC^1-schwierig [Rob93]. Daher stellt sich natürlicherweise die Frage, ob das Wortproblem einer Gruppe genau dann NC^1-schwierig ist, wenn die Gruppe nicht-auflösbar ist. Eine bejahende Antwort auf diese Frage erscheint sehr unwahrscheinlich. Allerdings kann man bei fast allen Gruppen, die in dieser Arbeit betrachtet werden, ein solches Verhalten beobachten: Alle betrachteten auflösbaren Gruppen haben Wortprobleme in TC^0 (nilpotente Gruppen [4], siehe Kapitel 5, Kranzprodukte [3], siehe Kapitel 4), während "natürliche" Beispiele nicht-auflösbarer Gruppen NC^1-schwierige Wortprobleme haben (nicht-auflösbare verallgemeinerte Baumslag-Solitar-Gruppen [5], siehe Kapitel 3, die Grigorchuk-Gruppe, die Thompson-Gruppen F, T und V, nicht-auflösbare lineare

[1] Eine Gruppe heißt endlich erzeugt, wenn es eine endliche Menge $\Sigma \subseteq G$ gibt, sodass sich jedes Element in G als ein Wort über dem Alphabet Σ ausdrücken lässt. In dieser Arbeit sind alle aufgeführten Gruppen endlich erzeugt – auch wenn dies nicht explizit erwähnt wird.

[2] NC^1 sind die Sprachen, die von Schaltkreisen logarithmischer Tiefe mit konstantem Eingangsgrad erkannt werden.

[3] ACC^0 (bzw. $\mathsf{ACC}^0[p]$) ist die Klasse der von Schaltkreisen konstanter Tiefe, polynomieller Größe und unbeschränktem Eingangsgrad mit Boolschen und Modulo-Gattern (bzw. Modulo-p-Gattern) akzeptierten Sprachen.

[4] TC^0 ist wie ACC^0 definiert, aber anstelle von Modulo-Gattern dürfen *Threshold*-Gatter verwendet werden, die den Wert 1 zurückgeben, sobald die Anzahl an Einsen im Input einen gewissen Wert übersteigen. Es gilt $\mathsf{ACC}^0 \subseteq \mathsf{TC}^0 \subseteq \mathsf{NC}^1$ – ob die Teilmengenbeziehung echt ist, ist nicht bekannt.

[5] Numerische Referenzen gehören zu Veröffentlichungen, die hier zusammengefasst werden, während alphanumerische Referenzen auf andere Veröffentlichungen verweisen.

Gruppen [1], siehe Kapitel 7, für Definitionen siehe Kapitel 2). Die einzige Ausnahme bildet ein Beispiel einer nicht-auflösbaren Gruppe, die in [1] konstruiert wurde. Von dieser Gruppe ist es unbekannt, ob das Wortproblem NC^1-schwierig ist; tatsächlich scheint dies eher unwahrscheinlich – zumindest falls $\mathsf{NC}^1 \neq \mathsf{TC}^0$ gilt.

Die obigen Beobachtungen setzen die Komplexität von Wortproblemen von Gruppen in Bezug zur Schaltkreiskomplexität – insbesondere zu der Frage, ob $\mathsf{TC}^0 \neq \mathsf{NC}^1$ gilt. Dieser Zusammenhang wird auch in [3] näher diskutiert. Insbesondere bietet sich hier die Möglichkeit, diese Schaltkreisklassen auch durch gruppentheoretische Probleme besser zu verstehen. Eine Lösung der Frage, ob $\mathsf{TC}^0 \neq \mathsf{NC}^1$ gilt, ist jedoch außer Reichweite (insbesondere, da nicht einmal bekannt ist, ob $\mathsf{TC}^0 \neq \mathsf{NP}$ gilt). Der Zusammenhang zwischen NC^1-Schwierigkeit und Nicht-Auflösbarkeit wird auch in [1] thematisiert.

Weitere algorithmische Probleme der Gruppentheorie. Außer dem Wortproblem gibt es zahlreiche weitere algorithmische Probleme in der Gruppentheorie. Die folgenden Probleme werden dabei in der vorliegenden Zusammenfassung behandelt. Hierbei sei G eine Gruppe mit endlichem Erzeugendensystem Σ.

- Das *Konjugationsproblem* $\mathrm{CP}(G)$: Gegeben Wörter $v, w \in \Sigma^*$, ist v konjugiert zu w in G (d. h. gibt es ein $z \in G$ mit $z^{-1}vz = w$ in G[6])?
- Das *Untergruppenproblem*: Gegeben Wörter $v_1, \dots, v_n \in \Sigma^*$ und $w \in \Sigma^*$, ist w in der Untergruppe $H = \langle v_1, \dots, v_n \rangle$ enthalten?

Das Konjugationsproblem ist neben dem Wortproblem auch eines von Dehns drei fundamentalen Problemen [Deh11] (das dritte ist das *Isomorphieproblem*, das hier nicht betrachtet wird). Ebenso wie das Untergruppenproblem ist das Konjugationsproblem eine natürliche Verallgemeinerung des Wortproblems, denn ein Element ist genau dann konjugiert zum neutralen Element einer Gruppe, wenn es schon selbst das neutrale Element ist. Das Konjugationsproblem ist von besonderem Interesse wegen seinen potentiellen Anwendungen in der nicht-kommutativen Kryptographie [CJ12, GS09, KLC$^+$00, SZ06, WWC$^+$11]. Hierbei wird die Beobachtung benutzt, dass es einfach ist, zwei konjugierte Elemente zu erzeugen (indem man ein Wort mit einem anderen Wort konjugiert), aber das Lösen des Konjugationsproblems trotzdem schwierig sein kann.

In einigen Fällen betrachten wir auch *uniforme* Varianten der genannten Probleme. Dies bedeutet, dass auch die Gruppe Teil der Eingabe ist. Um trotzdem noch entscheidbare Probleme zu erhalten, muss natürlich die Klasse der als Eingabe erlaubten Gruppen eingeschränkt sowie eine geeignete Kodierung der Gruppen gewählt werden. Details dazu werden in den jeweiligen Abschnitten beschrieben.

In den letzten Jahren kamen auch zunehmend Varianten von algorithmischen Problemen auf, bei denen die Eingaben in komprimierter Form gegeben sind. Hierzu werden folgende Varianten des Wortproblems untersucht:

- Das *komprimierte Wortproblem* $\textsc{CompressedWP}(G)$: Gegeben ein Straight-Line-Programm[7], das ein Wort $w \in \Sigma^*$ produziert, ist $w = 1$ in G?
- Ein Power-Wort ist ein Tupel $(p_1, x_1, p_2, x_2, \dots, p_n, x_n)$, wobei $p_i \in \Sigma^*$ und die x_i binär kodierte ganze Zahlen sind. Das *Power-Wortproblem* $\textsc{PowerWP}(G)$ ist die folgende Fragestellung: Gegeben ein Power-Wort $(p_1, x_1, p_2, x_2, \dots, p_n, x_n)$, ist $p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n} = 1$ in G?

[6] Während "$v = w$" die Gleichheit als Wörter bezeichnet, wird die Gleichheit in der Gruppe G durch "$v = w$ in G" oder "$v =_G w$" ausgedrückt.

[7] Ein Straight-Line-Programm ist eine kontextfreie Grammatik, die genau ein Wort produziert.

Klasse von Gruppen	WP	PowerWP [2]	CompressedWP
GBS-Gruppen	LOGSPACE[a)][b)] [5]	?	?
nilpotent	TC^0 [Rob93], [4][c)]	TC^0	DET, $\mathsf{C}_=\mathsf{L}$-schwierig [KL18a]
Grigorchuk-Gruppe G	LOGSPACE [GZ91], NC^1-schwierig [1]	LOGSPACE[d)]	PSPACE-vollst. [1]
Thompson-Gruppe F	LOGCFL [LS07], NC^1-schwierig [1]	?	PSPACE-vollst. [1]
nicht-abelsch frei	LOGSPACE [LZ77], NC^1-schwierig [Rob93]	LOGSPACE[a)]	P-vollst. [Loh06]
$G \wr \mathbb{Z}$ mit G abelsch	TC^0 [3]	TC^0	coRP [KL18a]
$G \wr \mathbb{Z}$ mit G endl. nicht-auflösbar	NC^1-vollst. [Waa90]	coNP-vollst.	PSPACE-vollst. [1]
$F_2 \wr \mathbb{Z}$	LOGSPACE[a)] [Waa90], NC^1-schwierig [Rob93]	coNP-vollst.	PSPACE-vollst. [1]

[a)] AC^0-Turing-reduzierbar auf WP(F_2).

[b)] Uniforme Variante – eine GBS-Gruppe in geeigneter Kodierung ist Teil der Eingabe.

[c)] Uniforme Variante – eine nilpotente Gruppe konstanter Klasse und Rangs ist Teil der Eingabe.

[d)] AC^0-many-one-reduzierbar auf WP(G).

Tabelle 1: Zusammenfassung der Resultate zum Wortproblem und seinen komprimierten Varianten. Hier bezeichnet F_2 die freie Gruppe vom Rang zwei. GBS-Gruppen sind verallgemeinerte Baumslag-Solitar-Gruppen wie siehe unten definiert.

Sowohl Straight-Line-Programme als auch Power-Wörter stellen natürliche Formen der Kompression dar und können als Verallgemeinerungen von Binärdarstellungen für ganze Zahlen angesehen werden. Das komprimierte Wortproblem ist ferner von Interesse wegen seiner Anwendung beim Lösen von (normalen) Wortproblemen in Automorphismengruppen und bestimmten semidirekten Produkten [Loh14]. Das Power-Wortproblem bildet eine natürliche Zwischenstufe zwischen dem normalen Wortproblem und dem komprimierten Wortproblem, denn einerseits kann auch eine exponentielle Komprimierung wie mit Straight-Line-Programmen erreicht werden, andererseits ist es in vielen Fällen aber tatsächlich deutlich einfacher zu lösen als das komprimierte Wortproblem. Die Resultate zu den komprimierten Varianten des Wortproblems sind in Tabelle 1 zusammengefasst. Hier kann man die beschriebenen Beobachtungen zu den unterschiedlichen Komplexitäten deutlich erkennen.

Liste der Publikationen. Die vorliegende Arbeit fasst die folgenden Veröffentlichungen zusammen (in chronologischer Reihenfolge aufgeführt):

- *A logspace solution to the word and conjugacy problem of generalized Baumslag-Solitar groups* [5].

- *The conjugacy problem in free solvable groups and wreath products of abelian groups is in* TC^0 [3]. Diese Arbeit ist eine gemeinsame Veröffentlichung mit Alexei Miasnikov und Svetla Vassileva. Eine Konferenzversion [MVW17] erschien bei der CSR 2017 und erhielt dafür einen Best-Paper-Award. Vorläufige Resultate wurden in [MVW18] veröffentlicht.

- TC^0 *circuits for algorithmic problems in nilpotent groups* [4]. Dies ist eine gemeinsame Veröffentlichung mit Alexei Miasnikov.

- *The power word problem* [2]. Dies ist eine gemeinsame Arbeit mit Markus Lohrey.

- *Groups with* $\mathsf{ALOGTIME}$*-hard word problems and* PSPACE*-complete compressed word problems* [1]. Dies ist eine gemeinsame Veröffentlichung mit Laurent Bartholdi, Michael Figelius, and Markus Lohrey.

- *Hardness of equations over finite solvable groups under the exponential time hypothesis* [6].

Gemeinsam ist diesen Veröffentlichungen, dass sie sich mit der Einordnung der oben genannten gruppentheoretischen Probleme in die Landschaft der Komplexitätsklassen zwischen TC^0 und $\mathsf{LOGSPACE}$ beschäftigen (in einigen Fällen gibt es auch Schwierigkeits- bzw. Vollständigkeitsresultate für größere Klassen, siehe auch Tabelle 1). Die letzte Arbeit fällt in mehreren Aspekten etwas aus diesem Rahmen heraus: Es werden nur endliche Gruppen betrachtet, es wird die Lösbarkeit von Gleichungen untersucht (eine Fragestellung, die sonst keine Rolle spielt) und es werden (durch die Exponential Time Hypothesis bedingte) quasipolynomielle untere Zeitschranken bewiesen. Es gibt trotzdem zwei wichtige Zusammenhänge zu den übrigen Veröffentlichungen: Erstens kann man auch hier die Auswirkung der Auflösbarkeit einer Gruppe auf die Komplexität von algorithmischen Problemen beobachten und zweitens basiert der verwendete Beweis auf einer effizienten Kodierung der UND-Funktion in eine Gruppe, was auch in der Schaltkreiskomplexität eine wichtige Rolle spielt.

Im Folgenden werden die einzelnen Veröffentlichungen thematisch eingeordnet und die jeweils wichtigsten Resultate zusammengefasst. Eine Übersicht über die Anteile des Autors an den jeweiligen Veröffentlichungen wird in den einzelnen Kapiteln der ausführlichen, englischsprachigen Zusammenfassung gegeben.

Um jegliche Copyrightverletzungen zu vermeiden sind die Originalartikel hier nicht abgedruckt. Stattdessen sind in der Publikationsliste Links sowohl zu den Originalpublikationen angegeben als auch zu im Wesentlichen inhaltsgleichen technischen Berichten auf `arxiv.org`.

A logspace solution to the word and conjugacy problem of generalized Baumslag-Solitar groups [5]. Baumslag-Solitar-Gruppen sind Gruppen der Form $\mathbf{BS}_{p,q} = \langle\, a, y \mid ya^p y^{-1} = a^q \,\rangle$ für ganze Zahlen p und q. Sie wurden 1962 von Baumslag und Solitar [BS62] als Beispiel für endlich präsentierte, nicht-Hopfsche Gruppen eingeführt. Baumslag-Solitar-Gruppen können als HNN-Erweiterung einer unendlichen zyklischen Gruppe geschrieben werden und passen damit in das allgemeinere Konzept von Fundamentalgruppen endlicher Graphen von Gruppen wie von Serre eingeführt [Ser80]. Als solche ergibt sich eine natürliche Verallgemeinerung zu Fundamentalgruppen endlicher Graphen von Gruppen mit unendlichen zyklischen Knoten- und Kantengruppen. Solche Gruppen werden auch *verallgemeinerte Baumslag-Solitar*-Gruppen (GBS-Gruppen) genannt und wurden schon häufiger in der Literatur betrachtet [Bee11, For03, Kro90].

Die Untersuchung algorithmischer Probleme in GBS-Gruppen hat eine recht lange Geschichte. Während die Entscheidbarkeit des Wortproblems einfach zu zeigen ist, gab es für das Konjugationsproblem, angefangen mit der Entscheidbarkeit des Konjugationsproblems in normalen Baumslag-Solitar-Gruppen [AS74], mehrere partielle Resultate [Ans76a, Ans76b, Ans76c, Hor84, HF94], bis schließlich Lockhart [Loc92] und Beeker [Bee11] unabhängig voneinander die Entscheidbarkeit des Konjugationsproblems in beliebigen GBS-Gruppen bewiesen.

Erste Schranken an die Komplexität des Wortproblems in einigen Spezialfällen ergaben sich durch das allgemeine Resultat zum Wortproblem linearer Gruppen in $\mathsf{LOGSPACE}$ [LZ77, Sim79].

Dies trifft insbesondere auf die Gruppen $\mathbf{BS}_{1,q}$ sowie $\mathbf{BS}_{q,\pm q}$ mit $q \in \mathbb{Z}$ zu. Später untersuchte Waack [Waa81] die GBS-Gruppe $\langle\, a, s, t \mid sas^{-1} = a, tat^{-1} = a^2 \,\rangle$ als Beispiel einer nicht-linearen Gruppe mit einem Wortproblem in $\mathsf{LOGSPACE}$. Ferner zeigte Robinson [Rob93], dass die Wortprobleme von Baumslag-Solitar-Gruppen $\mathbf{BS}_{1,q}$ mit $q \in \mathbb{Z}$ in TC^0 sind. Die Baumslag-Solitar-Gruppen $\mathbf{BS}_{1,q}$ entsprechen gerade den auflösbaren GBS-Gruppen. Da nicht-auflösbare GBS-Gruppen stets eine (nicht-abelsche) freie Untergruppe enthalten und eine solche ein NC^1-schwieriges Wortproblem hat [Rob93], können wir andererseits nicht erwarten, dass in diesem Fall das Wort- oder Konjugationsproblem in TC^0 lösbar ist. In der Dissertation des Autors wurde $\mathsf{LOGDCFL}$[8] als eine erste Komplexitätsschranke für das Wort- und Konjugationsproblem beliebiger GBS-Gruppen gezeigt.

Die hier zusammengefasste Arbeit [5] verbessert dieses Ergebnis nochmals, indem gezeigt wird, dass das Wortproblem beliebiger GBS-Gruppen auf das Wortproblem einer (nicht-abelschen) freien Gruppe reduziert werden kann. Hierbei wird eine TC^0-many-one-Reduktion verwendet. Insbesondere ergibt sich auch eine AC^0-Turing-Reduktion auf das Wortproblem einer (nicht-abelschen) freien Gruppe. Ferner wird gezeigt, dass das Problem, eine Britton-reduzierte Normalform zu berechnen, ebenfalls mittels einer AC^0-Turing-Reduktion auf das Wortproblem einer (nicht-abelschen) freien Gruppe reduziert werden kann. Dies wird verwendet, um eine Reduktion des gleichen Typs vom Konjugationsproblem einer beliebigen GBS-Gruppe auf das Wortproblem einer (nicht-abelschen) freien Gruppe zu beschreiben. Insbesondere kann sowohl das Wort- als auch Konjugationsproblem in $\mathsf{LOGSPACE}$ gelöst werden.

Bei den bisher genannten Resultaten ist die jeweilige GBS-Gruppe fest gewählt und *nicht* Teil der Eingabe. Betrachtet man dagegen die uniforme Variante, bei der eine beliebige GBS-Gruppe Teil der Eingabe ist, ergibt sich ein interessantes Phänomen: Während das Wortproblem immer noch in $\mathsf{LOGSPACE}$ ist (und bei einer geeigneten Kodierung auch immer noch AC^0-Turing-reduzierbar auf das Wortproblem einer (nicht-abelschen) freien Gruppe), wird das Konjugationsproblem $\mathsf{EXPSPACE}$-vollständig. Dies hängt damit zusammen, dass das Wortproblem für kommutative Monoide in das Konjugationsproblem von GBS-Gruppen kodiert werden kann. Für ein festes kommutatives Monoid ist das Wortproblem in NC^1 lösbar [IJCR91], aber sobald das kommutative Monoid Teil der Eingabe ist, wird das Problem $\mathsf{EXPSPACE}$-vollständig [CLM76, MM82].

The conjugacy problem in free solvable groups and wreath products of abelian groups is in TC^0 [3]. Das Konjugationsproblem in Kranzprodukten wurde erstmals 1966 von Matthews untersucht: $A \wr B$ hat genau dann ein entscheidbares Konjugationsproblem, wenn sowohl A als auch B ein entscheidbares Konjugationsproblem hat und das sogenannte Power-Problem in B berechenbar ist [Mat66].[9] Als Folge ergab sich, dass das Konjugationsproblem in frei meta-abelschen Gruppen entscheidbar ist. Kargapolov und Remeslennikov verallgemeinerten dies zu frei auflösbaren Gruppen von beliebigem Grad [KR66]. Einige Jahre später gelang es Remeslennikov und Sokolov [RS70] auch Matthews' Ergebnisse auf iterierte Kranzprodukte zu verallgemeinern, indem sie das Power-Problem in diesen Gruppen lösten. Außerdem zeigten sie, dass die Magnus-Einbettung [Mag39] von frei auflösbaren Gruppen in iterierte Kranzprodukte frei abelscher Gruppen Konjugiertheit (und Nicht-Konjugiertheit) erhält, und gaben damit einen neuen Beweis für die Entscheidbarkeit des Konjugationsproblems in frei auflösbaren Gruppen.

[8] $\mathsf{LOGDCFL}$ bedeutet in $\mathsf{LOGSPACE}$ reduzierbar auf eine deterministisch kontextfreie Sprache.

[9] Tatsächlich verwendete Matthews eine Reduktion auf das Zyklische-Untergruppen-Problem (Definition siehe unten), bezeichnete dieses aber als "Power-Problem". Da beide Probleme in Bezug auf Entscheidbarkeit äquivalent sind, ist diese Unterscheidung in diesem Fall aber nicht von Belang.

In [Vas11] zeigte Vassileva schließlich, dass Matthews' Kriterium für Konjugation in iterierten Kranzprodukten frei abelscher Gruppen in Polynomialzeit überprüft werden kann.

Als wichtiges Resultat zum Wortproblem ist noch folgendes Ergebnis von Myasnikov, Roman'kov, Ushakov und Vershik hervorzuheben: Das Wortproblem in frei auflösbaren Gruppen ist in Zeit $\mathcal{O}(n^3)$ lösbar – unabhängig vom Auflösbarkeitsgrad der Gruppe [MRUV10].

In der hier zusammengefassten Arbeit wird zunächst die Komplexität des Wortproblems genauer untersucht und gezeigt, dass das Wortproblem in $A \wr B$ AC^0-Turing-reduzierbar auf die Wortprobleme von A und B ist. Dies ist im Prinzip eine Wiederholung von [Waa90], wo allerdings nur eine NC^1-Reduktion beschrieben wird.

Um die Komplexität des Konjugationsproblems in Kranzprodukten zu charakterisieren, werden noch noch drei weitere, nah miteinander verwandte Probleme eingeführt:

- Das *Zyklische-Untergruppen-Problem* $\mathrm{CSGMP}(G)$: Gegeben $v, w \in \Sigma^*$, ist $w \in \langle v \rangle$ (d. h. gibt es ein $k \in \mathbb{Z}$ mit $v^k =_G w$)?

- Das *Zyklisches-Untermonoid-Problem* $\mathrm{CSMMP}(G)$: Gegeben $v, w \in \Sigma^*$, ist w in dem von v erzeugten Untermonoid (d. h. gibt es ein $k \in \mathbb{N}$ mit $v^k =_G w$)?

- Das *Power-Problem* $\mathrm{PP}(G)$: Gegeben $v, w \in \Sigma^*$, entscheide, ob es ein $k \in \mathbb{Z}$ gibt mit $v^k =_G w$ und, wenn ja, berechne die Binärdarstellung von k.

Insbesondere ist das Power-Problem eine Berechnungsvariante des Zyklische-Untergruppen-Problems. Außerdem sind diese Probleme in Bezug auf Entscheidbarkeit alle äquivalent, in Bezug auf Komplexität ist dies allerdings nicht klar; es gilt aber folgende Kette von Reduktionen $\mathrm{WP}(G) \leq_{\mathrm{m}}^{\mathsf{AC}^0} \mathrm{CSGMP}(G) \leq_{T}^{\mathsf{AC}^0} \mathrm{CSMMP}(G) \leq_{T}^{\mathsf{AC}^0} \mathrm{PP}(G)$. Das zentrale Resultat zum Konjugationsproblem ist wie folgt:

- $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{PP}(B))$,[10]

- $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSMMP}(B))$, falls B torsionsfrei ist,

- $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSGMP}(B))$, falls A abelsch ist.

Der Beweis hierzu ist eine detaillierte Auswertung von Matthews' Konjugationskriterium [Mat66]. Als Korollar ergibt sich, dass das Konjugationsproblem im rechts-iterierten Kranzprodukt $A \wr^d B$ abelscher Gruppen in TC^0 ist. Umgekehrt wird gezeigt, dass $\mathrm{CSGMP}(B) \leq_{\mathrm{m}}^{\mathsf{AC}^0} \mathrm{CP}(A \wr B)$, sobald A nicht-trivial ist. Falls A nicht-abelsch ist, gilt sogar $\mathrm{CSMMP}(B) \leq_{\mathrm{m}}^{\mathsf{AC}^0} \mathrm{CP}(A \wr B)$. Damit sind der zweite und dritte Fall des Hauptresultats in jedem Fall optimal. Eine AC^0- oder TC^0-Reduktion des Power-Problems von B auf $\mathrm{CP}(A \wr B)$ konnte dagegen nicht gezeigt werden und erscheint auch eher unwahrscheinlich. Es bleibt daher die Frage offen, ob auch schon im Allgemeinen $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSMMP}(B))$ gilt.

Um auch das Konjugationsproblem in links-iterierten Kranzprodukten zu lösen, wird außerdem noch folgendes Transfer-Resultat für das Power-Problem gezeigt: Sei $\beta \in \mathbb{N}$ und sei die Ordnung jedes Torsionselements in A β-glatt (d.h. nur Primteiler $\leq \beta$). Dann gilt $\mathrm{PP}(A \wr B) \in \mathsf{TC}^0(\mathrm{PP}(A), \mathrm{PP}(B))$. Daraus ergibt sich als Korollar, dass das Konjugationsproblem im links-iterierten Kranzprodukt $A\, {}^{d}\!\wr B$ in TC^0 ist, wenn A und B abelsch sind oder auflösbare Baumslag-Solitar-Gruppen. Insbesondere ist das Konjugationsproblem jeder frei auflösbaren Gruppe in TC^0.

[10] $\mathsf{TC}^0(L_1, \dots, L_k)$ bezeichnet eine TC^0-Turing-Reduktion auf die Sprachen $L_1, \dots, L_k$, d. h. TC^0-Schaltkreise, die auch Orakel-Gatter für $L_1, \dots, L_k$ verwenden dürfen.

TC^0 **circuits for algorithmic problems in nilpotent groups** [4]. In der Klasse der endlich erzeugten nilpotenten Gruppen sind viele algorithmische Probleme effizient entscheidbar (mit einigen Ausnahmen wie das Erfüllbarkeitsproblem von Gleichungen, siehe z.B. [GMO20]). Die Entscheidbarkeit des Wort- und Untergruppenproblems wurde schon 1958 von Mal'cev bewiesen [Mal58]. Wenige Jahre später zeigte Blackburn auch die Entscheidbarkeit des Konjugationsproblems. Erste Komplexitätsresultate für das Wortproblem ergaben sich durch Lipton und Zalcsteins Ergebnisse für lineare Gruppen [LZ77]. Diese wurden von Robinson [Rob93] auf TC^0 verbessert.

Ein typischer algorithmischer Ansatz, um mit nilpotenten Gruppen zu arbeiten, ist die Verwendung sogenannter Mal'cev-Basen (siehe z.B. [Hal69, KM79]). Diese erlauben, die Multiplikation in der Gruppe durch die Auswertung von Polynomen zu beschreiben (siehe Lemma 5.1), und wurden systematisch z.B. in [KRR$^+$69] und [Mos66] eingesetzt.

In [MMNV15] gaben Macdonald, Miasnikov, Nikolaev und Vassileva $\mathsf{LOGSPACE}$-Algorithmen für zahlreiche algorithmische Probleme in nilpotenten Gruppen an – unter anderem für das Wortproblem, das Untergruppenproblem und das Konjugationsproblem. Dabei betrachteten sie auch eine uniforme Variante der Probleme – allerdings wird die Zahl der Erzeugenden und die Nilpotenzklasse der Eingabegruppe als konstant angesehen.

Im Folgenden bezeichne $\mathcal{N}_{c,r}$ die Klasse der nilpotenten Gruppen mit r Erzeugern und von Klasse maximal c. Für die meisten der in [MMNV15] betrachteten Probleme konnte die genaue Komplexität in der hier zusammengefassten Arbeit [4] als TC^0 bestimmt werden. Insbesondere wird folgendes gezeigt:

- Das Wort-, Untergruppen- und Konjugationsproblem sowie einige weitere Probleme für nilpotente Gruppen sind in TC^0. Hierbei ist die Gruppe $G \in \mathcal{N}_{c,r}$ Teil der Eingabe, aber c und r sind konstant. Da alle diese Probleme TC^0-schwierig sind, ist ihre Komplexität damit exakt bestimmt.

- Unär kodierte Systeme linearer Gleichungen über den ganzen Zahlen können in TC^0 gelöst werden, wenn die Anzahl der Gleichungen durch eine Konstante beschränkt ist.

- Um die vorigen Resultate zu zeigen, wird das Erweiterte-ggT-Problem betrachtet (gegeben $a_1, \dots, a_n \in \mathbb{Z}$, berechne $x_1, \dots, x_n \in \mathbb{Z}$ mit $\mathrm{ggT}(a_1, \dots, a_n) = \sum_i a_i x_i$): Bei unär kodierter Eingabe kann dies in TC^0 gelöst werden.

- Es werden auch Varianten der Probleme betrachtet, bei welchen die Eingabe durch Wörter mit binären Exponenten gegeben ist. Dies sind Wörter der Form $p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n}$, wobei die p_i Elemente eines festen endlichen Erzeugendensystems der Gruppe und die x_i binär kodierte ganze Zahlen sind (und damit Spezialfälle von Power-Wörtern):
  - Das Wortproblem mit binär kodierten Exponenten ist auch in TC^0 (allgemeiner gilt dies sogar für das Power-Wortproblem, siehe Theorem 6.1).
  - Die anderen der betrachteten Probleme können mittels TC^0-Turing-Reduktionen auf das Erweiterte-ggT-Problem mit binär kodierter Eingabe reduziert werden. Für letzteres Problem ist nicht einmal ein NC-Algorithmus bekannt (tatsächlich hat der aktuell beste parallele Algorithmus eine Laufzeit von $\mathcal{O}(n/\log n)$ [Sed17]). Obwohl keine Reduktion des Erweiterte-ggT-Problems auf das Untergruppenproblem von nilpotenten Gruppen bekannt ist, scheint es eher unwahrscheinlich, dass letzteres gelöst werden kann, ohne das Erweiterte-ggT-Problem zu lösen.

The power word problem [2]. Wie bereits erwähnt, ist das Power-Wortproblem eine spezielle Variante des komprimierten Wortproblems. Explizit wurde das Power-Wortproblem erstmals in der hier zusammengefassten Arbeit [2] eingeführt. Allerdings wurde diese Variante des Wortproblems schon davor in mehreren Arbeiten untersucht ohne explizit so genannt zu werden. Insbesondere sind hier die Arbeiten [Ge93] zu algebraischen Zahlkörpern und [GS07] zum Untergruppenproblem in freien Gruppen zu nennen. Wie oben beschreiben, wurde außerdem der Spezialfall, dass die p_i Wörter der Länge eins sind, in [4] für nilpotente Gruppen untersucht. Allerdings ergeben sich hier auch durch Power-Wörter keine zusätzlichen Schwierigkeiten (siehe unten).

Ein nah verwandtes Problem ist das Rucksack-Problem über einer Gruppe G mit Erzeugendensystem Σ (siehe [GKLZ18, LZ18, MNU15]): Auf Eingabe von $w, w_1, \ldots, w_n \in \Sigma^*$, ist es die Frage, ob es $x_1, \ldots, x_n \in \mathbb{Z}$ gibt, sodass $w = w_1^{x_1} \cdots w_n^{x_n}$ in G gilt. In der kürzlich erschienenen Arbeit [FGLZ20] untersuchen Figelius, Ganardi, Lohrey und Zetzsche den Zusammenhang zwischen dem Power-Wortproblem und dem Rucksack-Problem. Unter anderem zeigen sie, dass das Power-Wortproblem in iterierten Kranzprodukten frei abelscher Gruppen in TC^0 ist, wohingegen das Power-Wortproblem in $G \wr \mathbb{Z}$ mit G uniform SENS (für eine Definition, siehe unten bzw. in [1]) coNP-schwierig ist (insbesondere erweitern sie damit Theorem 6.6 dieser Arbeit). Mithilfe dieser Ergebnisse wird in [FGLZ20] weiter gezeigt, dass das Rucksack-Problem in iterierten Kranzprodukten frei abelscher Gruppen NP-vollständig ist und für Gruppen der Form $G \wr \mathbb{Z}$ mit G uniform SENS sogar Σ_p^2-schwierig[11].

In der hier zusammengefassten Arbeit [2] werden folgende Resultate bewiesen:

- Ist G eine endlich erzeugte nilpotente Gruppe, dann ist $\textsc{PowerWP}(G)$ in TC^0. Der Beweis dieses Theorems ist eine einfache Schlussfolgerung aus [4] (siehe Lemma 5.1 und Theorem 5.2 in Kapitel 5).

- Das zentrale Ergebnis der Arbeit ist Theorem 6.2: Das Power-Wortproblem einer freien Gruppe ist AC^0-Turing-reduzierbar auf das Wortproblem der freien Gruppe mit zwei Erzeugenden.

  Der Beweis dieses Theorems beruht auf der Beobachtung, dass die Perioden p_i eines Power-Worts so modifiziert werden können, dass sie eindeutig sind im folgenden Sinne: kürzen sich "ausreichend lange" Faktoren von $p_1^{x_1}$ und $p_2^{x_2}$, dann gilt schon $p_1 = p_2$. Dies kann benutzt werden, um die Größe der Exponenten zu reduzieren und damit eine Reduktion auf das Wortproblem der freien Gruppe zu geben.

- Sei G endlich erzeugt und $H \leq G$ von endlichem Index. Dann ist das Power-Wortproblem von G NC^1-many-one-reduzierbar auf das Power-Wortproblem von H. Als Korollar daraus ergibt sich, dass das Power-Wortproblem virtuell freier Gruppen AC^0-Turing-reduzierbar auf das Wortproblem der freien Gruppe mit zwei Erzeugenden ist.

- Das Power-Wortproblem von Gruppen der Form $G \wr \mathbb{Z}$ mit G abelsch ist in TC^0.

- Das Power-Wortproblem von Gruppen der Form $G \wr \mathbb{Z}$ mit G endlich nicht-auflösbar oder G (nicht-abelsch) frei ist coNP-vollständig.

- Das Power-Wortproblem der Grigorchuk-Gruppe ist AC^0-many-one-reduzierbar auf ihr Wortproblem und damit insbesondere in $\mathsf{LOGSPACE}$.

[11] Σ_p^2 bezeichnet hier das zweite Level der Polynomialzeithierarchie, d. h. $\Sigma_p^2 = \exists\forall\mathsf{P}$.

Groups with $\mathsf{ALOGTIME}$-hard word problems and PSPACE-complete compressed word problems [1]. Wie bereits oben erwähnt zeigte Barrington [Bar89] einen erstaunlichen Zusammenhang zwischen dem Wortproblem von Gruppen und der Komplexitätstheorie: Für jede endliche nicht-auflösbare Gruppe G ist $\mathrm{WP}(G)$ vollständig für $\mathsf{ALOGTIME}$ (= $\mathsf{DLOGTIME}$-uniformes NC^1). Ferner ist die Reduktion denkbar einfach: Jedes Output-Bit hängt von maximal einem Input-Bit ab. Daher kann man sagen, dass $\mathsf{ALOGTIME}$ vollständig durch Gruppentheorie charakterisiert werden kann.

Barringtons Konstruktion basiert auf der Beobachtung, dass die UND-Funktion durch Kommutatoren simuliert werden kann (wie auch in der Arbeit [6] zu Gleichungen verwendet – siehe Kapitel 8). Dies erklärt den Zusammenhang zwischen Komplexität und Auflösbarkeit der Gruppe. Insbesondere erscheint damit einleuchtend, dass das Wortproblem endlicher p-Gruppen in $\mathsf{ACC}^0[p]$ und damit nicht $\mathsf{ALOGTIME}$-schwierig ist: Diese sind nilpotent und somit sind alle iterierten Kommutatoren genügend großer Tiefe trivial.

Robinsons NC^1-Schwierigkeits-Resultat [Rob93] für das Wortproblem freier Gruppen basiert auf einer ähnlichen Kommutatortechnik wie Barringtons Konstruktion. Der erste Beitrag der hier zusammengefassten Arbeit ist, die Barringtons und Robinsons Beweisen zugrundeliegenden Ideen zu abstrahieren. Dafür werden sogenannte *uniform stark effizient nicht-auflösbare* (kurz: *uniform SENS*) Gruppen eingeführt. Eine Gruppe G ist uniform SENS, falls es eine Konstante $\mu \in \mathbb{N}$ sowie Wörter $g_{d,v} \in \Sigma^*$ für alle $d \in \mathbb{N}$, $v \in \{0,1\}^{\leq d}$ gibt, sodass gilt:

(a) $|g_{d,v}| = 2^{\mu d}$ für alle $v \in \{0,1\}^d$,

(b) $g_{d,v} = [g_{d,v0},\, g_{d,v1}]$ für alle $v \in \{0,1\}^{<d}$,

(c) $g_{d,\varepsilon} \neq 1$ in G, und

(d) gegeben $v \in \{0,1\}^d$, eine binär kodierte positive ganze Zahl i mit μd Bits und $a \in \Sigma$, kann man in Linearzeit entscheiden, ob der i-te Buchstabe von $g_{d,v}$ ein a ist.

In einer SENS Gruppe gibt es daher effizient berechenbare Zeugen der Nicht-Auflösbarkeit (d.h. effizient berechenbare balancierte iterierte Kommutatoren beliebiger Tiefe). Die Klasse der SENS Gruppen erfüllt zahlreiche wünschenswerte Eigenschaften. Insbesondere ist die Definition unabhängig vom Erzeugendensystem Σ. Ferner ist G uniform SENS, sobald eine Untergruppe oder ein Quotient von G uniform SENS ist.

Nach Barringtons Resultat sind alle endlichen nicht-auflösbaren Gruppen uniform SENS. Wegen der oben genannten Abschlusseigenschaften sind damit auch (nicht-abelsche) freie Gruppen uniform SENS (was auch direkt aus Robinsons Argumenten folgt).

Dem Beweis Barringtons folgend wird in [1] als erstes wichtiges Resultat gezeigt, dass das Wortproblem von uniform SENS Gruppen $\mathsf{ALOGTIME}$-schwierig ist (bzw. NC^1-schwierig). Etwas salopp ausgedrückt bedeutet dies: Das Wortproblem jeder nicht-auflösbaren Gruppe G ist $\mathsf{ALOGTIME}$-schwierig – außer die Zeugen für die Nicht-Auflösbarkeit werden sehr schnell zu lange oder sind nicht effizient berechenbar. Tatsächlich wird in [1] auch ein Beispiel einer nicht-auflösbaren Gruppe präsentiert, die nicht uniform SENS ist.

Weitere Beispiele für uniform SENS Gruppen sind Gruppen G, für die gilt $G \wr H \leq G$ für eine nicht-triviale Gruppe H. Dies wird benutzt, um die uniform-SENS-Eigenschaft für die Thompson-Gruppen, die Grigorchuk-Gruppe und zahlreiche weitere sogenannte weakly branched Gruppen nachzuweisen (für Definitionen siehe Kapitel 2). Insbesondere ist das Wortproblem aller dieser Gruppen NC^1-schwierig, was zusammen mit dem $\mathsf{LOGSPACE}$-Algorithmus für kontrahierende Automatengruppen eine recht genaue Klassifizierung der Komplexität des Wortproblems der

Grigorchuk-Gruppe und einiger weiterer Gruppen liefert. Da die gleiche Einordnung auch für die Komplexität des Wortproblems von (nicht-abelschen) freien Gruppen gilt, stellt sich als offenes Problem die Frage, ob das Wortproblem einer freien Gruppe auf das der Grigorchuk-Gruppe reduziert werden kann (oder umgekehrt).

Eine weitere Anwendung der uniform-SENS-Definition ist folgende Dichotomie für lineare Gruppen: Das Wortproblem einer linearen Gruppe ist entweder in TC^0 oder $\mathsf{ALOGTIME}$-schwierig. Der Beweis hierfür basiert einerseits auf Tits' Alternative [Tit72], dass jede lineare Gruppe entweder eine freie Untergruppe enthält oder virtuell auflösbar ist, und andererseits auf [KL18a, Theorem 7], wo König und Lohrey zeigen, dass das Wortproblem auflösbarer linearer Gruppen in TC^0 ist.

Der zweite Teil der Arbeit dreht sich um das komprimierte Wortproblem CompressedWP. Für endliche Gruppen wurde dies in [BMPT97] untersucht und eine ähnliche Dichotomie wie für das (normale) Wortproblem gezeigt: Für endliche auflösbare Gruppen ist CompressedWP(G) in NC^2 – für endliche nicht-auflösbare Gruppen dagegen P-vollständig (der Beweis basiert ebenfalls auf einer Kommutatortechnik ähnlich wie für SENS Gruppen). Für das komprimierte Wortproblem in Kranzprodukten $G \wr \mathbb{Z}$ mit G abelsch beschrieben König und Lohrey einen coRP-Algorithmus [KL18b]. Dagegen ist CompressedWP($G \wr \mathbb{Z}$) coNP-schwierig [Loh14], wenn G nicht-abelsch ist.

Das Hauptresultat der hier zusammengefassten Arbeit zum komprimierten Wortproblem ist wie folgt: Sei G eine nicht-triviale Gruppe mit Zentrum $Z(G)$. Dann gilt:

- CompressedWP($G \wr \mathbb{Z}$) ist in $\forall\mathsf{LEAF}(\mathrm{WP}(G))$, und
- CompressedWP($G \wr \mathbb{Z}$) ist schwierig für $\forall\mathsf{LEAF}(\mathrm{WP}(G/Z(G)))$.

Insbesondere ist CompressedWP($G \wr \mathbb{Z}$) vollständig für $\forall\mathsf{LEAF}(\mathrm{WP}(G))$, falls $Z(G) = 1$ ist. Für eine Sprache L bezeichnet $\mathsf{LEAF}(L)$ die durch Blattsprachen definierte Komplexitätsklasse, wie in [BCS92, Her97, HLS+93, HVW96, JMT96] untersucht. Als Beispiele ergeben sich, dass CompressedWP($G \wr \mathbb{Z}$) vollständig ist für $\forall\mathsf{MOD}_p\mathsf{P}$, wenn G eine nicht-abelsche endliche p-Gruppe ist. Ferner ist CompressedWP($S_3 \wr \mathbb{Z}$) vollständig für $\forall\mathsf{MOD}_3{\oplus}\mathsf{P}$.

Obiges Resultat zu CompressedWP($G \wr \mathbb{Z}$) kann nun zusammen mit der SENS-Definition angewendet werden. Daraus folgt dann, dass das komprimierte Wortproblem in Kranzprodukten $G \wr \mathbb{Z}$ mit G endlich nicht-auflösbar oder G (nicht-abelsch) frei PSPACE-vollständig ist. Gleiches gilt für das komprimierte Wortproblem von Thompson-Gruppen, der Grigorchuk-Gruppe und zahlreichen weiteren Gruppen. Insbesondere sind dies die ersten Beispiele "natürlicher" Gruppen, deren komprimiertes Wortproblem beweisbar schwieriger ist als ihr Wortproblem (ein Beispiel einer speziell konstruierten Gruppe mit dieser Eigenschaft ist schon in [WW20] zu finden).

Hardness of equations over finite solvable groups under the exponential time hypothesis [6]. Die Untersuchung von Gleichungen über algebraischen Strukturen hat eine lange Geschichte in der Mathematik. Gleichungen in endlichen Gruppen wurden dagegen erst in den letzten 25 Jahren näher betrachtet. Ein Grund hierfür ist wohl, dass die Lösbarkeit von Gleichungen über einer endlichen Gruppe trivialerweise entscheidbar ist – im Gegensatz zu unendlichen Gruppen, wo es zahlreiche Unentscheidbarkeitsresultate gibt (siehe z.B. [GMO20, Rom79]) aber auch entscheidbare Spezialfälle (siehe z.B. [Mak84, DE17, LS06]).

Sei G eine feste Gruppe und $\mathcal{X}$ eine Menge an Variablen. Das *Erfüllbarkeitsproblem* EQN-SAT(G) erhält als Eingabe einen *Ausdruck* $\alpha \in (G \cup \mathcal{X} \cup \mathcal{X}^{-1})^*$; die Frage ist, ob es eine Belegung $\sigma : \mathcal{X} \to G$ der Variablen gibt, sodass $\sigma(\alpha) = 1$ (hierbei wird die Definition von σ in kanonischer Weise auf Ausdrücke ausgeweitet). Analog wird das *Identitätsproblem* EQN-ID(G) definiert. Hier ist die Fragestellung, ob *alle* Belegungen $\sigma(\alpha) = 1$ erfüllen.

In dieser Arbeit werden nur endliche Gruppen betrachtet. In diesem Fall kann man durch einen einfachen Guess-and-Check-Ansatz sehen, dass das Erfüllbarkeitsproblem in NP und das Identitätsproblem in coNP ist. Ferner ist das Erfüllbarkeitsproblem tendenziell schwieriger, da das Identitätsproblem in Polynomialzeit mittels Turing-Reduktionen auf das Erfüllbarkeitsproblem reduzierbar ist. Die genaue Komplexität ist jedoch ein aktives Forschungsthema, das durch Goldmann und Russell [GR02] initiiert wurde, indem sie zeigten, dass EQN-SAT(G) NP-vollständig ist, falls G nicht-auflösbar ist, und dass EQN-SAT für endliche nilpotente Gruppen in P ist. Ferner bewiesen sie für die Lösbarkeit von Gleichungs*systemen* die folgende Dichotomie: Für abelsche Gruppen können Gleichungssysteme in P gelöst werden, für nicht-abelsche Gruppen ist das Problem NP-vollständig.

Der Fall einer einzelnen Gleichung blieb allerdings für auflösbare aber nicht nilpotente Gruppen offen. In der Tat stellten Burris und Lawrence die Frage, ob EQN-ID$(G) \in$ P für alle endlichen auflösbaren Gruppen G gilt [BL04, Problem 1]. Ferner vermutete Horváth eine positive Antwort auf diese Frage [Hor11].

In der folgenden Zeit erschienen zahlreiche weitere Resultate zu Gleichungen in endlichen Gruppen. In [HLMS07] zeigten Horváth, Lawrence, Mérai und Szabó, dass auch EQN-ID(G) coNP-vollständig ist, falls G nicht-auflösbar ist. In [BL04, Hor15, HS06, Föl17] wurden für einige Klassen von nicht nilpotenten Gruppen Polynomialzeitalgorithmen für EQN-SAT und EQN-ID beschrieben. Das neueste Ergebnis in dieser Folge ist [FH20]: Ist G ein semidirektes Produkt einer p-Gruppe und einer abelschen Gruppe, dann ist EQN-SAT(G) in P und für semidirekte Produkte von nilpotenten und abelschen Gruppen ist EQN-ID in P. Eine Gemeinsamkeit all dieser Polynomialzeitalgorithmen ist, dass sie nur Gruppen mit Fittinglänge maximal zwei behandeln.[12] In der hier zusammengefassten Arbeit wird gezeigt, dass dies kein Zufall ist, sondern für kompliziertere Gruppen tatsächlich untere Schranken bewiesen werden können.

Die hier bewiesenen Schranken basieren auf der *Exponential Time Hypothesis (ETH)*. Diese Vermutung aus der Komplexitätstheorie impliziert, dass es keinen deterministischen Algorithmus für 3SAT gibt mit Laufzeit in $2^{o(n+m)}$, wobei n die Zahl der Variablen und m die Zahl der Klauseln der Eingabe ist[13].

Das Hauptresultat dieser Arbeit ist wie folgt: Sei G eine endliche auflösbare Gruppe und sei entweder

- die Fittinglänge von G mindestens vier oder
- die Fittinglänge von G drei und G habe keinen Normalteiler von Fittinglänge zwei und einer Zweierpotenz als Index.

Dann impliziert ETH, dass sowohl EQN-SAT(G) als auch EQN-ID(G) nicht in Polynomialzeit lösbar sind. Insbesondere wird auf Burris und Lawrence Fragestellung [BL04, Problem 1] unter ETH eine negative Antwort gegeben.

Tatsächlich ist die zusätzliche Bedingung im Fall von Fittinglänge drei nicht notwendig: In [IKK20] gaben Idziak, Kawałek und Krzaczkowski eine analoge untere Schranke für EQN-SAT(S_4) – in einer neuen gemeinsamen (nach der Einreichung dieser Habilitationsschrift erstellten) Arbeit [IKKW20], werden die unteren Schranken für alle Gruppen von Fittinglänge drei gezeigt.

[12] Die Fittinglänge einer Gruppe G ist das kleinste d, sodass es eine Folge $1 = G_0 \trianglelefteq \cdots \trianglelefteq G_d = G$ von Normalteilern gibt, bei der alle Quotienten G_{i+1}/G_i nilpotent sind.

[13] Beachte: Tatsächlich ist die Exponential Time Hypothesis eine noch stärkere Annahme.

In der hier zusammengefassten Arbeit [6] werden auch kurz die Konsequenzen für das Erfüllbarkeitsproblem in Halbgruppen betrachtet. Hier kann gezeigt werden, dass, sobald eine Halbgruppe einen Divisor hat, der die Bedingungen des Hauptresultats erfüllt, das Erfüllbarkeitsproblem in dieser Halbgruppe nicht in P ist unter ETH.

Der Beweis des Hauptresultats dieser Arbeit verwendet eine Reduktion des C-Färbbarkeitsproblems[14] für Graphen auf EQN-SAT. Dabei wird ein Graph mit m Kanten auf eine Gleichung der Länge $2^{\mathcal{O}(\sqrt{m})}$ abgebildet. Angenommen EQN-SAT könnte in Polynomialzeit gelöst werden, dann würde dies zu einem Algorithmus für das C-Färbbarkeitsproblem mit Laufzeit in $2^{\mathcal{O}(\sqrt{m})}$ führen, was im Widerspruch zu ETH steht.

Ein weiterer Aspekt, der in [6] untersucht wird, ist der Zusammenhang zwischen der sogenannten AND-Weakness-Vermutung und EQN-SAT: Um die Färbbarkeitsbedingung für einen Graphen in eine Gleichung über einer Gruppe zu kodieren, muss die UND-Funktion mit beliebig vielen Inputs in die Gruppe kodiert werden (an *keiner* Kante darf die Färbbarkeitsbedingung verletzt sein). Die AND-Weakness-Vermutung besagt, dass dies in auflösbaren Gruppen nur in exponentieller Größe geht. Tatsächlich impliziert die AND-Weakness-Vermutung nach [BMM$^+$00, Theorem 2], dass EQN-SAT in quasipolynomieller Zeit lösbar ist und damit die unteren Schranken des Hauptresultats im Wesentlichen scharf sind. Allerdings kann mit derselben Beweistechnik basierend auf iterierten Kommutatoren gezeigt werden, dass die AND-Weakness-Vermutung nicht zutrifft, wenn man "exponentiell" als $2^{\Omega(n)}$ liest. Eine Interpretation von "exponentiell" als $2^{n^{\Omega(1)}}$ ist dagegen konsistent zu den Ergebnissen dieser Arbeit sowie zu den früheren Schranken [BBR94].

[14] Eingabe hierfür ist ein Graph (V, E), und die Frage ist, ob eine Funktion $\chi : V \to \{1, \dots, C\}$ existiert, sodass $\chi(u) \neq \chi(v)$ für alle $\{u, v\} \in E$.

1 Introduction

Algorithmic problems in group theory have a long tradition, going back to the work of Dehn from 1911 [Deh11]. One of the most fundamental group-theoretic decision problems is the *word problem* for a group G with a finite[15] generating set Σ: does a given word $w \in \Sigma^*$ evaluate to the group identity? Novikov [Nov55] and Boone [Boo59] independently showed in the 1950's that there are finitely presented groups with undecidable word problems. On the other hand, in many important classes of groups the word problem is (efficiently) decidable. Famous examples are finitely generated linear groups, where the word problem belongs to deterministic logarithmic space ($\mathsf{LOGSPACE}$ for short) [LZ77, Sim79], and hyperbolic groups where the word problem can be solved in linear time [Hol00] as well as in LOGCFL [Loh05][16].

There are also several results concerning groups having word problems in classes even smaller than $\mathsf{LOGSPACE}$. Most notably are the results on finite groups: if a finite group is solvable, its word problem is in the circuit class ACC^0 [BT88][17]. Contrarily, for non-solvable finite groups the word problem is NC^1-complete[18] [Bar89].

However, not only finite groups can have easy word problems: Robinson [Rob93] showed that all nilpotent groups have word problems in TC^0.[19] This has been generalized to all solvable linear groups: their word problems are in TC^0 by the work of König and Lohrey [KL18a]. Moreover, the class of groups having word problems in TC^0 is closed under wreath products.[20]

On the other hand, no non-solvable group is known to have a word problem in TC^0. Furthermore, the word problem of a non-abelian free group is also hard for NC^1 [Rob93]. Thus, there is a natural question whether there is a dichotomy in a sense that non-solvable groups have NC^1-hard word problems while solvable groups do not.[21] Indeed in all papers summarized in this Habilitationsschrift we see such a behavior: all solvable groups we consider have a word problem in TC^0 (nilpotent groups [4], wreath products [3]), while "natural" examples of non-solvable groups have NC^1-hard word problems: non-solvable generalized Baumslag-Solitar groups[22] [5], the Grigorchuk group, Thompson's groups[23], and non-solvable linear groups [1] – though in [1] a non-solvable group is constructed which we cannot prove to have an NC^1-hard word problem. We explore this connection between non-solvability and complexity explicitly in [1] (Chapter 7 of this summary).

[15] In this work all groups are finitely generated – even if not explicitly mentioned.

[16] LOGCFL means $\mathsf{LOGSPACE}$-reducible to a context-free language.

[17] ACC^0 are the languages accepted by constant-depth, polynomial-size circuits with unbounded fan-in Boolean and modulo gates.

[18] NC^1 is the class of languages accepted by bounded fan-in Boolean circuits of logarithmic depth.

[19] TC^0 is the class of languages accepted by constant-depth, polynomial-size circuits with unbounded fan-in Boolean and majority gates.

[20] For the first explicit reference, see [MVW17] (summarized in Chapter 4) – in [Waa90] Waack used the same ideas to show that NC^1 is closed under wreath products).

[21] Notice that there are solvable groups with undecidable word problems [Rem73] – indeed, there are even finitely presented such examples [Kha81] Nevertheless, this does not prove NC^1-hardness since the reductions used in [Rem73, Kha81] might be too powerful.

[22] These comprise the Baumslag-Solitar groups $\left\langle a, y \;\middle|\; ya^py^{-1} = a^q \right\rangle$ for $p, q \in \mathbb{Z} \setminus \{0\}$ – for a definition see Chapter 3.

[23] For definitions see Chapter 2.

Notice that NC^1-completeness results are only known for finite non-solvable groups (more precisely, finite non-solvable extensions of groups with word problems in TC^0). On the contrary, there are several results on infinite non-solvable groups with word problems as difficult[24] as the word problem of a free group. Most importantly, this applies to right-angled Artin groups[25] [Kau17] and – more generally – to all subgroups of these and to graph products of groups whose word problems are AC^0-Turing-reducible to the word problem of a free group. In [5] we extend this class further to generalized Baumslag-Solitar groups. Moreover, the Grigorchuk group as well as Thompson's group F are other candidates for belonging to this class – although neither membership nor hardness results are known so far.

While the word problem is without any doubt the most fundamental algorithmic problem in group theory, there are also other important problems. We will introduce several of them in Chapter 2 as well as in the summaries of the individual papers which are part of this work. At this point we confine ourselves to describing a few more problems. For the *conjugacy problem* of a group G the input consists of two words over the generators of G and the question is whether the respective group elements are conjugate[26]. Clearly, the word problem is a special case of the conjugacy problem since an element is the identity if and only if it is conjugate to the identity. However, there are examples of groups with a decidable word problem but undecidable conjugacy problem [Mil71]. In recent years, the study of the conjugacy problem became quite popular because of its possible applications in non-commutative cryptography; see for example [CJ12, GS09, KLC$^+$00, SZ06, WWC$^+$11]. These applications use the fact that it is easy to create elements which are conjugate, but it might be difficult to check whether two given elements are conjugate – even if the word problem is easy. This observation holds for example in polycyclic groups, which have a word problem in TC^0 [Rob93], but the conjugacy problem is not even known to be in NP.

Nevertheless, in all cases we consider, the conjugacy problem is (efficiently) decidable. Indeed, [5], [3], and [4] (see Chapters 3, 4, and 5 of this summary, respectively) describe reductions of the conjugacy problem to the word problem in several classes of groups. However, notice that in [5] we also consider the uniform conjugacy problem where a generalized Baumslag-Solitar group is part of the input. Here, the complexity changes dramatically: the problem becomes $\mathsf{EXPSPACE}$-complete.

In recent years, compressed variants of the word problem gained an increasingly important role as well. Most prominently, there is the *compressed word problem:* the input consists of a *straight-line program*[27] and the question is whether the produced word evaluates to the identity of the group. Of course, in terms of decidability the compressed word problem and the word problem are equivalent, but in terms of complexity there may be differences. Indeed, in [WW20] Wächter and the author presented the first example of a group whose compressed word problem is provably more difficult than the ordinary word problem. While [WW20] uses an on-the-purpose construction, in [1] (Chapter 7 of this summary) we give "natural" examples for such a behavior by showing that the famous Grigorchuk group has a word problem in $\mathsf{LOGSPACE}$ and that its compressed word problem is PSPACE-complete. Moreover, the same observation applies to certain wreath products.

From a group-theoretic point of view, the compressed word problem is interesting not only because it is a natural succinct version of the word problem, but also because several classical word problems can be efficiently reduced to compressed word problems. For instance, the

[24] Here, "as difficult" means equivalent under AC^0-Turing reductions.

[25] Also known as *graph groups* or *free partially commutative groups.*

[26] More formally: on input of u and v, the question is whether there exists some z with $z^{-1}uz = v$ in G.

[27] A context-free grammar that produces exactly one word.

word problem for a finitely generated subgroup of $\mathrm{Aut}(G)$ reduces in polynomial time to the compressed word problem for G [Loh14, Theorem 4.6]. Similar statements apply to certain group extensions [Loh14, Theorems 4.8 and 4.9]. This motivates the search for groups with compressed word problem decidable in polynomial time like nilpotent groups [KL18a, MMNV15], hyperbolic groups [HLS19] and virtually special groups [Loh14]. Moreover, for finite groups we again see a dichotomy between solvable and non-solvable groups: for finite solvable groups the compressed word problem is in NC^2, whereas for finite non-solvable groups it is P-complete [BMPT97].

In this work we also consider other forms of compressed inputs, namely *power words* ([2], see Chapter 6) and *words with binary exponents* ([4], see Chapter 5). With such inputs, the complexity of the word problem is somewhere in between the complexity of the ordinary word problem and the complexity of the compressed word problem. Indeed, we shall see cases where the complexity of the power word problem and the ordinary word problem are (almost) the same, while in other classes of groups the power word problem is more difficult under standard assumptions from complexity theory.

This cumulative Habilitationsschrift comprises publications investigating the problems described above. The connection between these articles is the focus on small complexity classes defined by circuits. One common observation is the above-described relation between solvability of a group and the complexity of its word problem. The paper [6] (summarized in Chapter 8), which is on equations in finite solvable groups, is somehow different: technically, it does not apply circuit complexity. Nevertheless, it firstly sheds some more light on the interplay between the solvability of a group G and the complexity of algorithmic problems in G and secondly uses techniques that also give some upper bounds on circuits sizes (resp. G-program lengths) for computing the AND-function. The following publications are summarized in this cumulative Habilitationsschrift (in chronological order):

- *A logspace solution to the word and conjugacy problem of generalized Baumslag-Solitar groups* [5].

- *The conjugacy problem in free solvable groups and wreath products of abelian groups is in* TC^0 [3]. Joint work with Alexei Miasnikov and Svetla Vassileva. Conference version at CSR 2017 [MVW17] (best paper award). Preliminary results have been published in [MVW18].

- TC^0 *circuits for algorithmic problems in nilpotent groups* [4]. Joint work with Alexei Miasnikov.

- *The power word problem* [2]. Joint work with Markus Lohrey.

- *Groups with* $\mathsf{ALOGTIME}$*-hard word problems and* PSPACE*-complete compressed word problems* [1]. Joint work with Laurent Bartholdi, Michael Figelius, and Markus Lohrey.

- *Hardness of equations over finite solvable groups under the exponential time hypothesis* [6].

Outline. Chapter 2 introduces some definitions and clarifies the notation. The following chapters summarize the above-listed publications. Each chapter consists of a short review on the background of the respective paper followed by a summary of its results. In some of the chapters some details regarding the proofs are given, too. For papers with multiple authors, the results are also attributed to the individual authors.

Throughout the summary, numerical references (e. g. [5]) refer to one of the six summarized publications, whereas alphanumerical references (e. g. [WW20]) refer to other publications.

In order to avoid any copyright violation, the original articles are not part of this Habilitationsschrift. Instead in the publication list links to the original articles as well as to technical reports are given.

2 Preliminaries

Words. An *alphabet* is a (finite or infinite) set Σ; an element $a \in \Sigma$ is called a *letter*. The free monoid over Σ is denoted by Σ^*, its elements are called *words*. The multiplication of the monoid is the concatenation of words. The identity element is the empty word denoted by 1 or ε. The length of a word w is denoted by $|w|$.

Groups. We consider a group G together with a surjective homomorphism $\eta : \Sigma^* \to G$ (a *monoid presentation*) for some (finite or infinite) alphabet Σ. In order to keep notation simple, we suppress the homomorphism η and consider words also as group elements. We write $v =_G w$ or "$v = w$ in G" as a shorthand of $\eta(v) = \eta(w)$.

Often we will assume that Σ is a *standard generating set* meaning that $1 \in \Sigma$ and for every $a \in \Sigma$ also the inverse a^{-1} belongs to Σ. In this case we have a natural involution on Σ^* defined by $(a_1 \cdots a_n)^{-1} = a_n^{-1} \cdots a_1^{-1}$ for $a_i \in \Sigma$ (which is the same as forming inverses in the group).

Throughout, we only consider finitely generated groups – even if not explicitly mentioned.

Free groups. We denote the free group over some basis Λ by F_Λ. It can be defined taking all words over $\Sigma = \Lambda \cup \overline{\Lambda}$ (where $\overline{\Lambda}$ is a disjoint copy of Λ) modulo the relations $a\bar{a} = \bar{a}a = 1$ for $a \in \Lambda$. We write F_2 as shorthand for $F_{\{a,b\}}$.

Wreath products. Let G and H be groups. Consider the set $G^{(H)}$ of all mappings $f \colon H \to G$ such that the *support* $\operatorname{supp}(f) := \{h \in H \mid f(h) \neq 1\} \subseteq H$ is finite. It forms a group with pointwise multiplication. The group H has a natural left action on $G^{(H)}$ given by $hf(a) = f(h^{-1}a)$, where $f \in G^{(H)}$ and $h, a \in H$. The corresponding semidirect product $G^{(H)} \rtimes H$ is the (restricted) *wreath product* $G \wr H$. In other words:

- Elements of $G \wr H$ are pairs (f, h), where $h \in H$ and $f \in G^{(H)}$.
- The multiplication in $G \wr H$ is defined as follows: Let $(f_1, h_1), (f_2, h_2) \in G \wr H$. Then $(f_1, h_1)(f_2, h_2) = (f, h_1h_2)$, where $f(a) = f_1(a)f_2(h_1^{-1}a)$.

Commutators and nilpotent groups. We write $[x, y] = x^{-1}y^{-1}xy$ for the commutator and $x^y = y^{-1}xy$ for the conjugation. Moreover, we write $[x_1, \ldots, x_n] = [[x_1, \ldots, x_{n-1}], x_n]$ for $n \geq 3$. For subgroups $H_1, H_2 \leq G$, we write $[H_1, H_2] = \langle\{[h_1, h_2] \mid h_1 \in H_1, h_2 \in H_2\}\rangle$. A group G is called *nilpotent* if it has a finite central series, i.e.

$$G = G_1 \geq G_2 \geq \cdots \geq G_c \geq G_{c+1} = 1$$

such that $[G, G_i] \leq G_{i+1}$ for all $i = 1, \ldots, c$.

Richard Thompson's groups. In 1965 Richard Thompson introduced three finitely presented groups $F < T < V$ acting on the unit-interval, the unit-circle and the Cantor set, respectively.

A standard reference for Thompson's groups is [CFP96]. Of these three groups, F received most attention. It has a finite presentation with two generators:

$$F = \langle x_0, x_1 \mid [x_0x_1^{-1}, x_0^{-1}x_1x_0],\ [x_0x_1^{-1}, x_0^{-2}x_1x_0^2] \rangle.$$

The infinite presentation $F = \langle x_0, x_1, x_2, \dots \mid x_k^{x_i} = x_{k+1} \text{ for all } i < k \rangle$ is also very convenient. The group F is torsion-free, its derived subgroup $[F, F]$ is simple and the center of F is trivial. Moreover, by the following fact, F contains the limit group $H_\infty = \bigcup_{i \geq 0} H_i$, where $H_0 = \mathbb{Z}$ and $H_{i+1} = H_i \wr \mathbb{Z}$:

Lemma 2.1 ([GS99, Lemma 20])**.** *The group F contains a subgroup isomorphic to $F \wr \mathbb{Z}$.*

From a computational perspective it is interesting to note that all three Thompson's groups are co-context-free[28] [LS07]. This implies that the word problems for Thompson's groups are in LOGCFL.

Weakly branched groups. Let X be a finite set. The free monoid X^* serves as the vertex set of a regular rooted tree with an edge between v and vx for all $v \in X^*$ and all $x \in X$. The group W of automorphisms of this tree naturally acts on the set X of level-1 vertices, and permutes the subtrees hanging from them. Exploiting the bijection $X^+ = X^* \times X$, we thus have an isomorphism

$$\varphi \colon W \to W \wr \operatorname{Sym}(X) = W^X \rtimes \operatorname{Sym}(X) \tag{2.1}$$

mapping $g \in W$ to elements $f \in W^X$ and $\pi \in \operatorname{Sym}(X)$ as follows: π is the restriction of g to $X \subseteq X^*$, and f is uniquely defined by $(xv)^g = x^\pi v^{f(x)}$. We write $g@x$ for $f(x)$ and call it the *state (or coordinate) of g at x*. If $X = \{0, \dots, k\}$ we write $g = \langle\langle g@0, \dots, g@k \rangle\rangle \pi$. For more details, [BGv03, Nek05] serve as good references.

Definition 2.2. Let W and φ be as in the previous paragraph. A subgroup $G \leq W$ is *self-similar* if $\varphi(G) \leq G \wr \operatorname{Sym}(X)$. In other words: the actions on subtrees xX^* are given by elements of G itself. A self-similar group G is *weakly branched* if there exists a non-trivial subgroup $K \leq G$ with $\varphi(K) \geq K^X$. In other words: for every $k \in K$ and every $x \in X$ the element acting as k on the subtree xX^* and trivially elsewhere belongs to K. A subgroup K as above is called a *branching subgroup.*

There exist important examples of f.g. self-similar weakly branched groups, notably the *Grigorchuk group G*, see [Gri80]: It is generated by elements a, b, c, d, and acts on the rooted tree X^* for $X = \{0, 1\}$. The action, and therefore the whole group, are defined by the restriction of φ to G's generators: $\varphi(a) = (0, 1)$, $\varphi(b) = \langle\langle a, c \rangle\rangle$, $\varphi(c) = \langle\langle a, d \rangle\rangle$, and $\varphi(d) = \langle\langle 1, b \rangle\rangle$, where we use the notation $(0, 1)$ for the non-trivial element of $\operatorname{Sym}(X)$ (that permutes 0 and 1) and $\langle\langle w_0, w_1 \rangle\rangle$ for a tuple in $G^{\{0,1\}} \cong G \times G$. It is well-known that the Grigorchuk group G is infinite, torsion, weakly branched, and all its finite subquotients are 2-groups (so in particular nilpotent). Moreover, it has a f.g. branching subgroup.

Other examples of f.g. self-similar weakly branched groups with a f.g. branching subgroup include the Gupta-Sidki groups [GS83], the Hanoi tower groups [Gv06], and all iterated monodromy groups of degree-2 complex polynomials [BN08] except z^2 and $z^2 - 2$.

2.1 Complexity

We use the standard complexity classes LOGSPACE, P, NP, coNP, and PSPACE as defined in any complexity theory textbook (see e. g. [Pap94]).

[28] Meaning that the set of all words representing non-trivial group elements is a context-free language.

Random access Turing machines. Since we also deal with sublinear time complexity classes, we use Turing machines with *random access*. Such a machine has an additional index tape and some special query states. Whenever the Turing machine enters a query state, the following transition depends on the input symbol at the position which is currently written on the index tape in binary notation. We define the following complexity class $\mathsf{DLINTIME}$ (resp. $\mathsf{DLOGTIME}$) as the class of languages that can be accepted by a deterministic Turing machine in linear (resp. logarithmic) time. Moreover, a language is in $\mathsf{ALOGTIME}$ if it can be accepted by an alternating Turing machine in logarithmic time.

A function $f\colon \Gamma^* \to \Sigma^*$ is $\mathsf{DLOGTIME}$-computable if there is some polynomial p with $|f(x)| \leq p(|x|)$ for all $x \in \Gamma^*$ and the set $L_f = \{(x, a, i) \mid x \in \Gamma^* \text{ and the } i\text{-th letter of } f(x) \text{ is } a\}$ belongs to $\mathsf{DLOGTIME}$. Here i is a binary encoded integer. A $\mathsf{DLOGTIME}$-reduction is a $\mathsf{DLOGTIME}$-computable many-one reduction.

Circuit Complexity. The class AC^0 (resp. TC^0) is defined as the class of functions computed by families of circuits of constant depth and polynomial size with unbounded fan-in Boolean gates (AND, OR, NOT) (resp. unbounded fan-in Boolean and Majority gates) – the alphabets are encoded over the binary alphabet $\{0, 1\}$.

In the following, we only consider $\mathsf{DLOGTIME}$-uniform circuit families and we write AC^0 (resp. TC^0) as shorthand for $\mathsf{DLOGTIME}$-uniform AC^0 (resp. TC^0). Roughly speaking, a circuit family is called $\mathsf{DLOGTIME}$-uniform if the function mapping the string 1^n to some proper encoding of the n-input circuit is $\mathsf{DLOGTIME}$-computable in the above sense. It is well-known that $\mathsf{ALOGTIME} = \mathsf{DLOGTIME}$-uniform NC^1, see [Vol99] for details.

Reductions. Let $K \subseteq \Delta^*$ and $L \subseteq \Sigma^*$ and let $\mathcal{C}$ be a complexity class. Then K is *$\mathcal{C}$-many-one-reducible* to L (denoted by $K \leq_{\mathrm{m}}^{\mathcal{C}} L$) if there is a $\mathcal{C}$-computable function $f : \Delta^* \to \Sigma^*$ with $w \in K \iff f(w) \in L$.

A function f is *AC^0-(Turing)-reducible* to a function g if there is a $\mathsf{DLOGTIME}$-uniform family of AC^0 circuits computing f which, in addition to the Boolean gates, may also use oracle gates for g (i. e., gates which on input x output $g(x)$). This is expressed by $f \in \mathsf{AC}^0(g)$ or $f \leq_T^{\mathsf{AC}^0} g$. TC^0 (Turing) reducibility is defined analogously. We have the following inclusions (note that even $\mathsf{TC}^0 \subseteq \mathsf{P}$ is not known to be strict):

$$\mathsf{AC}^0 \subsetneq \mathsf{TC}^0 \subseteq \mathsf{AC}^0(F_2) \subseteq \mathsf{LOGSPACE} \subseteq \mathsf{P}.$$

2.2 Algorithmic problems in group theory

Let G be a group generated (as a monoid) by a finite set Σ. We define the following algorithmic problems:

Basic algorithmic problems. The following problems have a long history and have been widely studied:

- The *word problem*[29] $\mathrm{WP}(G)$: given a word $w \in \Sigma^*$, is $w =_G 1$? Alternatively we can view $\mathrm{WP}(G)$ as the formal language $\{w \in \Sigma^* \mid w =_G 1\}$.

- The *conjugacy problem* $\mathrm{CP}(G)$: given words $v, w \in \Sigma^*$, is v conjugate to w in G (i. e., is there some $z \in G$ with $v^z = w$ in G)?

[29] Sometimes we also refer to the word problem as *ordinary word problem* in order to emphasize the contrast to the compressed variants defined below.

- The *subgroup membership problem*: given $v_1, \dots, v_n \in \Sigma^*$ and $w \in \Sigma^*$, is w an element of the subgroup $H = \langle v_1, \dots, v_n \rangle$?

In some of the publications summarized in this treatise we also study so-called *uniform* versions of these problems. For the uniform version, also the ambient group is part of the input. Of course, if an arbitrary finitely presented group is part of the input, the uniform versions are all undecidable by [Nov55, Boo59]. Nevertheless, by restricting the input to certain classes of groups, these problems can still be decided efficiently. In particular, in Chapter 3 we consider the uniform word and conjugacy problem for generalized Baumslag-Solitar groups (definition see below) and in Chapter 5 we allow a nilpotent group of constant rank and nilpotency class to be part of the input.

Compressed variants of the word problem. In order to describe compressed variants of the word problem, we introduce the following methods for compression:

- A *straight-line program* (SLP) is a context-free grammar producing precisely one word. Without loss of generality, we can assume that this grammar is in Chomsky normal form. Thus, alternatively, we can think of an SLP with terminal alphabet Σ as a circuit over the monoid Σ^*: each gate is either an input gate labelled with an element of Σ or it computes the product of its two predecessor gates.
- A *power word* (over Σ) is a tuple $(p_1, x_1, p_2, x_2, \dots, p_n, x_n)$ where $p_1, \dots, p_n \in \Sigma^*$ are words over the group generators (called the periods of the power word) and $x_1, \dots, x_n \in \mathbb{Z}$ are integers that are encoded in binary. Such a power word represents the word $p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n}$.
- A *word with binary exponents* is a power word $(p_1, x_1, p_2, x_2, \dots, p_n, x_n)$ where $p_i \in \Sigma$ for all i.

In Chapter 5 we consider words with binary exponents as inputs for all the problems described in the previous paragraph. Using straight-line programs or power words we obtain two special variants of the word problem:

- The *compressed word problem* $\textsc{Compressed}\mathrm{WP}(G)$: given an SLP producing a word $w \in \Sigma^*$, is $w =_G 1$?
- The *power word problem* $\textsc{Power}\mathrm{WP}(G)$: given a power word $(p_1, x_1, p_2, x_2, \dots, p_n, x_n)$, is $p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n} =_G 1$?

Due to the binary encoded exponents, a power word can be seen as a succinct description of an ordinary word. Hence, a priori, the power word problem for a group G could be computationally more difficult than the word problem. Moreover, from a power word $(p_1, x_1, p_2, x_2, \dots, p_n, x_n)$ one can easily compute a straight-line program for the word $p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n}$. In this sense, the power word problem is at most as difficult as the compressed word problem. On the other hand, both power words and straight-line programs can achieve exponential compression; so the additional difficulty of the compressed word problem does not come from a higher compression rate but rather from straight-line programs generating “more complex” words.

3 A logspace solution to the word and conjugacy problem of generalized Baumslag-Solitar groups

3.1 Background

A *Baumslag-Solitar group* is a group of the form $\mathbf{BS}_{p,q} = \langle\, a, y \mid ya^py^{-1} = a^q \,\rangle$ where $p, q \in \mathbb{Z}\smallsetminus\{0\}$. These groups were introduced in 1962 by Baumslag and Solitar [BS62] as examples for finitely presented non-Hopfian two-generator groups.

The usual presentation of a Baumslag-Solitar group is an HNN extension of an infinite cyclic group with one stable letter. The different Baumslag-Solitar groups correspond to the different inclusions of the associated subgroup into the base group. HNN extensions are a special case of fundamental groups of graphs of groups – where the graph consists of exactly one vertex with one attached loop. Thus, there is a natural notion of a generalized Baumslag-Solitar group (GBS group) as a fundamental group of a graph of groups with infinite cyclic vertex and edge groups – see e. g. [Bee11, For03, Kro90].

It has been long known that both the word problem and the conjugacy problem in generalized Baumslag-Solitar groups are decidable. Actually, the standard application of Britton reductions leads to a polynomial time algorithm for the word problem (see e. g. [Lau12]). Decidability of the conjugacy problem has been shown by Anshel and Stebe for ordinary Baumslag-Solitar groups [AS74] and independently by Lockhart [Loc92] and Beeker [Bee11] for arbitrary GBS groups (see also [Ans76a, Ans76b, Ans76c, Hor84, HF94] for preliminary results).

First complexity bounds for particular cases of GBS groups are due to the result that linear groups have a word problem in LOGSPACE [LZ77, Sim79]. Later, Waack [Waa81] examined the particular GBS group $\langle\, a, s, t \mid sas^{-1} = a, tat^{-1} = a^2 \,\rangle$ as an example of a non-linear group which has a word problem in LOGSPACE. For solvable GBS groups (Baumslag-Solitar groups $\mathbf{BS}_{1,q}$ for $q \in \mathbb{Z}$) the word problem was shown to be in TC^0 by Robinson [Rob93]. Moreover, in [DMW16] it is shown that both the word and the conjugacy problem in $\mathbf{BS}_{1,2}$ are in uniform TC^0 (with a straightforward extension to $\mathbf{BS}_{1,q}$).

Since non-solvable GBS groups contain free groups, we cannot expect the word or conjugacy problem to be in TC^0 because of Robinson's lower bound [Rob93]. In the author's dissertation [Wei15], a LOGDCFL algorithm for the word problem of GBS groups has been given – which means that it is LOGSPACE-reducible to a deterministic context-free language.

This chapter summarizes the results obtained in [5]. Some preliminary results were part of the author's dissertation [Wei15].

3.2 Graphs of groups and GBS groups

Generalized Baumslag-Solitar groups are defined as fundamental groups of graphs of groups. Therefore, we give a brief introduction into this topic following [DW17], which in turn is based on Serre's book [Ser80]. A *graph* $Y = (V, E, \iota, \tau, \bar{\cdot})$ is given by a set of *vertices* $V = V(Y)$ and a set of *edges* $E = E(Y)$ together with two mappings $\iota, \tau : E \to V$ and an involution $e \mapsto \bar{e}$ without fixed points such that $\iota(e) = \tau(\bar{e})$.

Definition 3.1 (Graph of Groups)**.** Let $Y = (V(Y), E(Y))$ be a connected graph. A *graph of groups* $\mathcal{G}$ over Y is given by the following data:

(i) For each vertex $a \in V(Y)$, there is a *vertex group* G_a.

(ii) For each edge $y \in E(Y)$, there is an *edge group* G_y such that $G_y = G_{\overline{y}}$.

(iii) For each edge $y \in E(Y)$, there is an injective homomorphism from G_y to $G_{\iota(y)}$, which is denoted by $c \mapsto c^y$. The image of G_y in $G_{\iota(y)}$ is denoted by G_y^y.

In the following, Y is always a finite graph. Since $G_y = G_{\overline{y}}$, there is also a homomorphism $G_y \to G_{\tau(y)}$ with $c \mapsto c^{\overline{y}}$.

The fundamental group of $\mathcal{G}$ can be constructed as a subgroup of the larger group $F(\mathcal{G})$: as an (possibly infinite) alphabet we choose a disjoint union $\Delta = E(Y) \cup \bigcup_{a \in V(Y)} (G_a \setminus \{1\})$, and we define the group

$$F(\mathcal{G}) = \Delta^* \Big/ \left\{ gh = [gh],\, yc^{\overline{y}}\overline{y} = c^y \;\middle|\; a \in V(Y),\, g, h \in G_a;\, y \in E(Y),\, c \in G_y \right\},$$

where $[gh]$ denotes the element obtained by multiplying g and h in G_a (where $1 \in G_a$ is identified with the empty word). For $a, b \in V(Y)$ we define a subset $\Pi(\mathcal{G}, a, b) \subseteq \Delta^*$ by

$$\begin{aligned}\Pi(\mathcal{G}, a, b) = \{\, g_0 y_1 \cdots g_{n-1} y_n g_n \mid{}& y_i \in E(Y),\, \iota(y_1) = a,\, \tau(y_n) = b,\\ & \tau(y_i) = \iota(y_{i+1}),\, g_0 \in G_a,\, g_i \in G_{\tau(y_i)} \text{ for all } i \,\},\end{aligned}$$

where again $1 \in G_a$ is identified with the empty word. Moreover, we set $\Pi(\mathcal{G}) = \bigcup_{a \in V(Y)} \Pi(\mathcal{G}, a, a)$ and call $w = g_0 y_1 \cdots g_{n-1} y_n g_n \in \Pi(\mathcal{G})$ a *$\mathcal{G}$-factorization.*

There are two definitions of the *fundamental group* of a graph of groups:

(i) Let $a \in V(Y)$. The *fundamental group* $\pi_1(\mathcal{G}, a)$ of $\mathcal{G}$ with respect to the base point $a \in V(Y)$ is defined as the image of $\Pi(\mathcal{G}, a, a)$ in $F(\mathcal{G})$.

(ii) Let T be a spanning tree of Y (i. e., a subset of $E(Y)$ connecting all vertices and not containing any cycles). The *fundamental group* of $\mathcal{G}$ with respect to T is defined by

$$\pi_1(\mathcal{G}, T) = F(\mathcal{G}) \Big/ \{ y = 1 \mid y \in T \}.$$

By [Ser80, Proposition I.20], the canonical homomorphism from the subgroup $\pi_1(\mathcal{G}, a)$ of $F(\mathcal{G})$ to the quotient group $\pi_1(\mathcal{G}, T)$ is an isomorphism. In particular, the two definitions of the fundamental group are independent of the choice of the base point and the spanning tree.

Example 3.2. Let $\mathcal{G}$ be a graph of groups consisting of a single vertex a with a self-loop $\{y, \overline{y}\}$ and let $G_a = \mathbb{Z} = \langle a \rangle$ and $G_y = G_{\overline{y}} = \mathbb{Z} = \langle c \rangle$ and the inclusions given by $c^y = a^p$ and $c^{\overline{y}} = a^q$. for some $p, q \in \mathbb{Z} \setminus \{0\}$. Then the fundamental group $\pi_1(\mathcal{G}, a)$ is the Baumslag-Solitar group

$$\pi_1(\mathcal{G}, a) = \mathbf{BS}_{p,q} = \left\langle a, y \;\middle|\; ya^p y^{-1} = a^q \right\rangle.$$

Generalized Baumslag-Solitar Groups. A *generalized Baumslag-Solitar group* (*GBS group*) is a fundamental group of a finite graph of groups with only infinite cyclic vertex and edge groups. That means a GBS group G is completely given by a finite graph Y and numbers $\alpha_y, \beta_y \in \mathbb{Z} \setminus \{0\}$ for $y \in E(Y)$ such that $\alpha_y = \beta_{\overline{y}}$. For $a \in V(Y)$ we write $G_a = \langle a \rangle$. Then we have

$$F(\mathcal{G}) = \left\langle V(Y), E(Y) \;\middle|\; \overline{y}y = 1,\, yb^{\beta_y}\overline{y} = a^{\alpha_y} \text{ for } y \in E(Y),\, a = \iota(y),\, b = \tau(y) \right\rangle$$

and $G = \pi_1(\mathcal{G}, a) \leq F(\mathcal{G})$ for any $a \in V(Y)$.

3.3 Results

In this work, we show that both the word problem and the conjugacy problem of every generalized Baumslag-Solitar group are in $\mathsf{LOGSPACE}$. More precisely, we establish the following results:

Theorem 3.3. *Let G be a GBS group. There is a uniform TC^0 many-one reduction from the word problem of G to the word problem of the free group F_2.*

More precisely, let $G = \pi_1(\mathcal{G}, a) \cong \pi_1(\mathcal{G}, T)$ be a GBS group over a graph Y and denote $\Delta = E(Y) \cup \left\{a^k \;\middle|\; a \in V(Y),\, k \in \mathbb{Z}\right\}$. There is a uniform TC^0 many-one reduction from each of the problems

(i) *given a word $w \in \Delta^*$, decide whether w is a $\mathcal{G}$-factorization and, if so, decide whether $w =_{F(\mathcal{G})} 1$,*

(ii) *given a word $w \in \Delta^*$, decide whether $w =_{\pi_1(\mathcal{G},T)} 1$,*

to the word problem of the free group F_2. In particular, the word problem of G is in $\mathsf{LOGSPACE}$.

The proofs of these results rely in an essential way on the seminal work by Hesse that iterated multiplication can be done in TC^0 [Hes01, HAB02]. On the way for solving the conjugacy problem, we need to compute freely reduced words. Therefore, we establish the following result:

Proposition 3.4. *The following problem is AC^0-reducible to the word problem of F_2: given a finite alphabet Λ and a word $w \in (\Lambda \cup \overline{\Lambda})^*$, compute a freely reduced word $\hat{w} \in (\Lambda \cup \overline{\Lambda})^*$ with $\hat{w} =_{F_\Lambda} w$.*

Theorem 3.5. *Let G be a GBS group. The conjugacy problem of G is uniform-AC^0-Turing-reducible to the word problem of the free group. In particular, it is in $\mathsf{LOGSPACE}$.*

The proof of Theorem 3.5 uses Horadam's conjugacy criterion for graphs of groups [Hor81]. We apply a reduction to the word problem for a (fixed) finitely presented commutative semigroup, which can be solved in NC^1 by [IJCR91, Thm. 1].

Uniform versions of the word and conjugacy problem. We also consider uniform versions of the word and conjugacy problem where the GBS group is part of the input. For this we assume that the graph of groups is given in a proper encoding: we assume that the encoding consists of the numbers $|V(Y)|$ and $|E(Y)|$ and a list of tuples $(y, \iota(y), \tau(y), \alpha_y, \beta_y, \overline{y})$ for the edges. Here $y, \overline{y} \in \{0, \dots, |E(Y)| - 1\}$ and $\iota(y), \tau(y) \in \{0, \dots, |V(Y)| - 1\}$ and all numbers (also the α_y, β_y) are encoded as binary integers using the same number of bits for all y. Notice that the graph of groups also defines the alphabet $\Delta = E(Y) \cup \left\{a^k \;\middle|\; a \in V(Y),\, k \in \mathbb{Z}\right\}$.

We say an encoding is *valid*, if all tuples are properly formed, for every edge y, there is an inverse edge $\overline{y}$ satisfying $\iota(y) = \tau(\overline{y})$ and $\alpha_y = \beta_{\overline{y}}$, and the graph is connected.

Corollary 3.6. *The following problem is TC^0-many-one-reducible to the word problem of F_2. Input: a valid encoding of a graph of groups $\mathcal{G}$ and a word $w \in \Delta^*$. Decide whether w is a $\mathcal{G}$-factorization and, if so, decide whether $w =_{F(\mathcal{G})} 1$.*

Note that we need the promise in Corollary 3.6 that the input is a valid encoding of a graph of groups. Indeed, it cannot be checked whether the graph is connected in TC^0 unless $\mathsf{TC}^0 = \mathsf{LOGSPACE}$ (by [CM87], already connectivity for forests is $\mathsf{LOGSPACE}$-complete with respect to NC^1-reductions – the reduction is actually an AC^0 reduction, see also [JLM]). On the other hand, by [Rei08], connectivity of undirected graphs can be checked in $\mathsf{LOGSPACE}$. Hence, as the other points can be easily verified in AC^0, it can be checked in $\mathsf{LOGSPACE}$ whether an encoding of a graph of groups is valid.

The uniform version of Theorem 3.3 (ii) is not so immediate. The difficulty lies in the computation of the paths $T[a, b]$, a problem which is complete for $\mathsf{LOGSPACE}$.

Corollary 3.7. *The following problem is complete for* $\mathsf{LOGSPACE}$ *under* AC^0 *reductions: Given a (valid) encoding of graph of groups* $\mathcal{G}$ *with a spanning tree* T *(given as list of edges) and a word* $w \in \Delta^*$*. Decide whether* $w =_{\pi_1(\mathcal{G},T)} 1$*.*

Thus, the uniform version of the word problem for GBS groups is essentially as difficult as the word problem for a fixed GBS group. For conjugacy this picture changes dramatically. The *uniform conjugacy problem* for GBS groups receives as input a graph of groups $\mathcal{G}$ given as above and two $\mathcal{G}$-factorizations $v, w \in \Delta^*$. The question is whether v is conjugate to w in $F(\mathcal{G})$ (which is equivalent to conjugacy in the fundamental group with respect to a base point).

Theorem 3.8. *The uniform conjugacy problem for GBS groups is* $\mathsf{EXPSPACE}$*-complete – even if the numbers* α_y, β_y *are given in unary.*

The proof of Theorem 3.8 is a reduction from the uniform word problem for finitely presented commutative semigroups, which by [CLM76, MM82] is $\mathsf{EXPSPACE}$-complete.

4 The conjugacy problem in free solvable groups and wreath products of abelian groups is in TC^0

4.1 Background

The study of the conjugacy problem in wreath products has quite a long history: In 1966 Matthews proved that a wreath product $A \wr B$ has a decidable conjugacy problem if and only if both A and B have decidable conjugacy problems and B has a decidable *cyclic subgroup membership problem* [Mat66][30]. As a consequence, she obtained a solution to the conjugacy problem in free metabelian groups. Kargapolov and Remeslennikov generalized the result by establishing decidability of the conjugacy problem in free solvable groups of arbitrary degree [KR66].

A few years later Remeslennikov and Sokolov [RS70] also generalized Matthews' results to iterated wreath products by solving the cyclic subgroup membership problem in these groups. They also showed that the Magnus embedding [Mag39] of free solvable groups into iterated wreath products of abelian groups preserves conjugacy – thus, giving a new proof for decidability of the conjugacy problem in free solvable groups.

Later, Vassileva [Vas11] gave a polynomial time algorithm for the conjugacy problem in free solvable groups by showing that Matthews' conjugacy criterion [Mat66] for iterated wreath products of abelian groups can actually be checked in polynomial time.

4.2 Results

The word problem. The first result in [3] is a stronger version of [Waa90] where only NC^1 reducibility is shown:

Theorem 4.1. $\mathrm{WP}(A \wr B) \in \mathsf{AC}^0(\mathrm{WP}(A), \mathrm{WP}(B))$.

Definition 4.2. Let $d \in \mathbb{N}$. We define the *left-iterated wreath product*, $A\,{}^{d}\!\wr B$, and the *right-iterated wreath product* $A \wr^{d} B$ of two groups A and B inductively as follows:

- $A\,{}^{1}\!\wr B = A \wr B$
- $A\,{}^{d}\!\wr B = A \wr (A\,{}^{d-1}\!\wr B)$
- $A \wr^{1} B = A \wr B$
- $A \wr^{d} B = (A \wr^{d-1} B) \wr B$

Let $S_{d,r}$ denote the free solvable group of degree d and rank r. The Magnus embedding [Mag39] is an embedding $S_{d,r} \to \mathbb{Z}^r \wr S_{d-1,r}$. By iterating the construction, we obtain an embedding $S_{d,r} \to \mathbb{Z}^r\,{}^{d}\!\wr 1$. For the purpose of this paper, the explicit definition of the homomorphism is not relevant – it suffices to know that it is an embedding and that it preserves conjugacy [RS70]. The following corollary is also a consequence of [KLR07] since a wreath product can be embedded into the corresponding block product. It can be proved using the Magnus embedding.

[30] Actually, in [Mat66] the cyclic subgroup membership problem is called *power problem*.

Corollary 4.3. *Let A and B be finitely generated abelian groups and let $d \geq 1$. The word problems of $A \wr^d B$ and of $A \,{}^d\!\wr B$ are in TC^0. In particular, the word problem of a non-trivial free solvable group is TC^0-complete.*

Note that here the groups A, B and the number d of wreath products are fixed. Indeed, if there were a single TC^0 circuit which worked for free solvable groups of arbitrary degree, this circuit would also solve the word problem of the free group, which is NC^1-hard, – implying $\mathsf{TC}^0 = \mathsf{NC}^1$.

Open Problem 4.4. Can the word problem of a free solvable group of degree d be decided in TC^0 with majority depth less than d?

A negative answer to Open Problem 4.4 would imply a negative answer to the analog question for iterated block products (the converse is not clear). Moreover, we want to point out that Open Problem 4.4 is related to an important question in complexity theory: as outlined in [MT98], a negative answer would imply that $\mathsf{TC}^0 \neq \mathsf{NC}^1$. Nevertheless, the following observation points rather towards a positive answer to Open Problem 4.4: the word problem of free solvable groups is decidable in cubic time – regardless of the solvability degree d [MRUV10].

More algorithmic problems. In order to describe the complexity of the conjugacy problem in a wreath product precisely, we need to introduce some more algorithmic problems on groups. As before, let G be a group with finite generating set Σ.

- The *cyclic subgroup membership problem* CSGMP(G): given $v, w \in \Sigma^*$, is $w \in \langle v \rangle$ (i. e., is there some $k \in \mathbb{Z}$ with $v^k =_G w$)?

- The *cyclic submonoid membership problem* CSMMP(G): given $v, w \in \Sigma^*$, is w in the submonoid generated by v (i. e., is there some $k \in \mathbb{N}$ with $v^k =_G w$)?

- The *power problem* PP(G): given $v, w \in \Sigma^*$, decide whether there is some $k \in \mathbb{Z}$ such that $v^k =_G w$ and, in the "yes" case, compute this k in binary representation. If v has finite order in G, the computed k has to be the smallest non-negative such k.

Whereas the first two of these problems are decision problems, the last one is an actual computation problem – actually it can be seen as the computation variant of the cyclic subgroup membership problem. Be aware that sometimes in literature the power problem is defined as what we refer to as cyclic subgroup membership problem. We have

$$\mathrm{WP}(G) \leq_{\mathrm{m}}^{\mathsf{AC}^0} \mathrm{CSGMP}(G) \leq_{T}^{\mathsf{AC}^0} \mathrm{CSMMP}(G) \leq_{T}^{\mathsf{AC}^0} \mathrm{PP}(G).$$

In addition, notice that PP(G) can be reduced to CSGMP(G) via polynomial time Turing reductions using an iterated squaring approach, but an AC^0 reduction seems out of reach.

Example 4.5. Let $\mathbf{BS}_{1,2} = \langle\, a, t \mid tat^{-1} = a^2 \,\rangle$ be the Baumslag-Solitar group. The conjugacy problem of $\mathbf{BS}_{1,2}$ is in TC^0 by [DMW16]. Moreover, the power problem is also in TC^0. This can be seen by a straightforward calculation in the semidirect product description $\mathbb{Z}[1/2] \rtimes \mathbb{Z}$ of $\mathbf{BS}_{1,2}$. However, note that in this example the solution to the power problem can only be returned if encoded in binary because of the exponential distortion of the subgroup $\langle a \rangle$.

The conjugacy problem. The next result is the central result on conjugacy. It is the complexity version of the “if” part of [Mat66, Thm. B]. While Matthews only describes a reduction to the cyclic subgroup membership problem, we distinguish three cases with three (slightly) different reductions.

Theorem 4.6. *Let A and B be finitely generated groups. We have*

- $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{PP}(B))$*,*
- $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSMMP}(B))$ *if B is torsion-free,*
- $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSGMP}(B))$ *if A is abelian.*

Corollary 4.7. *Let A and B be finitely generated groups and $d \geq 1$. Then*

- $\mathrm{CP}(A \wr^d B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{PP}(B))$*,*
- $\mathrm{CP}(A \wr^d B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSMMP}(B))$ *if B is torsion-free.*

Notice that $A \wr^d B$ is not abelian (for non-trivial A and B). Hence, we *cannot* conclude that $\mathrm{CP}(A \wr^d B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSGMP}(B))$ even if A is abelian.

Corollary 4.8. *Let A and B be f. g. abelian groups and $d \geq 1$. Then* $\mathrm{CP}(A \wr^d B) \in \mathsf{TC}^0$*.*

The role of the power problem. The following result is a complexity analog of the “only if” part of [Mat66, Thm. B], which only considers decidability. Note that for pure decidability, it does not matter if we consider $\mathrm{CSGMP}(B)$, $\mathrm{CSMMP}(B)$ or $\mathrm{PP}(B)$ since they can all be reduced to each other. However, in terms of complexity in some cases we can show hardness results for $\mathrm{CSGMP}(B)$ and $\mathrm{CSMMP}(B)$.

Theorem 4.9. *Let A be finitely generated and non-trivial. Then* $\mathrm{CSGMP}(B) \leq_{\mathrm{m}}^{\mathsf{AC}^0} \mathrm{CP}(A \wr B)$*. If, moreover, A is non-abelian, then* $\mathrm{CSMMP}(B) \leq_{\mathrm{m}}^{\mathsf{AC}^0} \mathrm{CP}(A \wr B)$*.*

Theorem 4.9 shows that in the cases that A is abelian or B torsion-free Theorem 4.6 is the best possible result one could expect. However, it is not clear how $\mathrm{PP}(B)$ could possibly be reduced to $\mathrm{CP}(A \wr B)$ in TC^0. Thus, there remains the possibility that Theorem 4.6 could be strengthened in the general case.

Open Problem 4.10. Is $\mathrm{CP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CP}(A), \mathrm{CP}(B), \mathrm{CSMMP}(B))$ in general?

The power problem in iterated wreath products. In order to solve the conjugacy problem in left-iterated wreath products, we also need to solve the power problem in wreath products. In general, we do not know whether the power problem in a wreath product is in TC^0 given that the power problem of the factors is in TC^0. The issue is that when dealing with torsion it might be necessary to compute greatest common divisors – which is not known to be in TC^0. By restricting torsion elements to have only smooth orders, we circumvent this issue. A number is called β-smooth for some $\beta \in \mathbb{N}$ if it only contains prime factors less than or equal to β.

Theorem 4.11. *Let $\beta \in \mathbb{N}$ and suppose the order of every torsion element in A is β-smooth. Then we have* $\mathrm{PP}(A \wr B) \in \mathsf{TC}^0(\mathrm{PP}(A), \mathrm{PP}(B))$*.*

Roughly the proof of Theorem 4.11 works as follows: On input (b, f) and (c, g) first apply the power problem in B to b and c. If there is no solution, then there is also no solution for (b, f) and (c, g). Otherwise, the smallest $k \geq 0$ with $b^k =_B c$ can be computed. If b has infinite order, it remains to check whether $(b, f)^k = (c, g)$. Since k might be too large, this cannot be done by simply applying the word problem. Nevertheless, we only need to establish equality of functions in $A^{(B)}$. We show that it suffices to check equality on certain (polynomially many) "test points". In the case that b has finite order K, we know that if there is a solution to the power problem it must be in $k + K\mathbb{Z}$. Now, similar techniques as in the infinite order case can be applied to find the solution.

Corollary 4.12. *Let A and B be finitely generated abelian groups or Baumslag-Solitar groups $\mathbf{BS}_{1,q}$ for some $q \in \mathbb{Z} \setminus \{0\}$ and let $d \geq 1$. The conjugacy problem of $A \,{}^d\!\wr B$ is in TC^0. Also, the conjugacy problem of free solvable groups is in TC^0.*

Remark 4.13. In the arXiv version of [4] we show that also nilpotent groups have power problem and conjugacy problem in TC^0 and that the orders of torsion elements are uniformly bounded. Thus, also iterated wreath products of nilpotent groups have conjugacy problem in TC^0.

We have seen that in torsion-free groups it suffices to solve the cyclic submonoid membership problem instead of the power problem for deciding conjugacy. Therefore, it would be interesting to reduce the cyclic submonoid membership problem in a wreath product to the same problem in its factors.

Open Problem 4.14. Is $\mathrm{CSMMP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CSMMP}(A), \mathrm{CSMMP}(B))$ or similarly is $\mathrm{CSGMP}(A \wr B) \in \mathsf{TC}^0(\mathrm{CSGMP}(A), \mathrm{CSGMP}(B))$?

4.3 Contribution by the author

This chapter summarizes [3], which is joint work with Alexei Miasnikov and Svetla Vassileva. A conference version appeared at CSR 2017 [MVW17] (best paper award). Preliminary results have been published in [MVW18].

The preliminary results [MVW18] are mainly due to Svetla Vassileva and Alexei Miasnikov. The author made some minor contributions concerning the presentation and fixing some small mistakes. On the other hand, the extensions made in the conference paper [MVW17] and its corresponding journal version [3] over [MVW18] are in large parts due to the author. Most importantly, this comprises the discussion on the relation between the complexity of the word problem in wreath products and the question whether $\mathsf{TC}^0 = \mathsf{NC}^1$, the improvement from $\mathsf{LOGSPACE}$ to TC^0 for the word and conjugacy problem, the reduction of CSMMP and CSGMP to the conjugacy problem (Theorem 4.9), and the more refined analysis of the power problem in iterated wreath products (Theorem 4.11).

5 TC^0 circuits for algorithmic problems in nilpotent groups

5.1 Background

Finitely generated nilpotent groups are a class where many algorithmic problems are (efficiently) decidable (with some exceptions like the problem of solving equations – see e. g. [GMO20]). In 1958, Mal'cev [Mal58] established decidability of the word and subgroup membership problem by investigating finite approximations of nilpotent groups. In 1965, Blackburn [Bla65] showed decidability of the conjugacy problem. However, these methods did not allow any efficient (e. g. polynomial time) algorithms. Nevertheless, in 1966 Mostowski provided "practical" algorithms for the word problem and several other problems [Mos66]. A major step in terms of complexity was Lipton and Zalcstein's LOGSPACE algorithm [LZ77] for the word problem of linear groups. Together with the fact that finitely generated nilpotent groups are linear (see e. g. [Hal69, KM79]) this gives a LOGSPACE solution to the word problem of nilpotent groups, which was later improved to uniform TC^0 by Robinson [Rob93]. A typical algorithmic approach to nilpotent groups is using so-called Mal'cev (or Hall–Mal'cev) bases (see e. g. [Hal69, KM79]), which allow to carry out group operations by evaluating polynomials (see Lemma 5.1). This approach was systematically used in [KRR$^+$69] and [Mos66]. Finally, in [MMNV15] the following problems were shown to be in LOGSPACE using the Mal'cev basis approach. Here, $\mathcal{N}_{c,r}$ denotes the set of nilpotent groups which are of nilpotency class at most c and generated by at most r elements.

- The *(uniform) word problem*: given $G \in \mathcal{N}_{c,r}$ and $g \in G$, is $g = 1$ in G?
- given $G \in \mathcal{N}_{c,r}$ and $g \in G$, compute the (Mal'cev) normal form of g.
- The *subgroup membership problem*: given $G \in \mathcal{N}_{c,r}$ and $g, h_1, \dots, h_n \in G$, decide whether $g \in \langle h_1, \dots, h_n \rangle$ and, if so, express g as a word over the subgroup generators $h_1, \dots, h_n$ (in [MMNV15] only the decision version was shown to be in LOGSPACE – for expressing g as a word over the original subgroup generators a polynomial time bound was given).
- Given $G, H \in \mathcal{N}_{c,r}$ and $K = \langle g_1, \dots, g_n \rangle \leq G$, together with a homomorphism $\varphi : K \to H$ specified by $\varphi(g_i) = h_i$, and some $h \in \mathrm{Im}(\varphi)$, compute a generating set for $\ker(\varphi)$ and find $g \in G$ such that $\varphi(g) = h$.
- Given $G \in \mathcal{N}_{c,r}$ and $K = \langle g_1, \dots, g_n \rangle \leq G$, compute a presentation for K.
- Given $G \in \mathcal{N}_{c,r}$ and $g \in G$, compute a generating set for the centralizer of g.
- The *conjugacy problem*: given $G \in \mathcal{N}_{c,r}$ and $g, h \in G$, decide whether or not there exists $u \in G$ such that $u^{-1}gu = h$ and if so find such an element u.

Notice that these problems are not only of interest by themselves, but also might serve as building blocks for solving the same problems in polycyclic groups – which are of particular interest because of their possible application in non-commutative cryptography [EK04].

5.2 Preliminaries on nilpotent groups

Mal'cev coordinates. Let $G = G_1 \geq G_2 \geq \cdots \geq G_c \geq G_{c+1} = 1$ be a central series of G. If G is finitely generated, so are the abelian quotients G_i/G_{i+1}, $1 \leq i \leq c$. Let $a_{i1}, \ldots, a_{im_i}$ be a basis of G_i/G_{i+1}, i.e. a generating set such that G_i/G_{i+1} has a presentation $\left\langle a_{i1}, \ldots, a_{im_i} \,\middle|\, a_{ij}^{e_{ij}}, [a_{ik}, a_{i\ell}], \text{ for } j \in \mathcal{T}_i,\ k, \ell \in \{1, \ldots, m_i\} \right\rangle$, where $\mathcal{T}_i \subseteq \{1, \ldots, m_i\}$ (here $\mathcal{T}$ stands for torsion) and $e_{ij} \in \mathbb{Z}_{>0}$ (be aware that we explicitly allow $e_{ij} = 1$, which is necessary for our definition of quotient presentations in Section 5.2). Formally, we put $e_{ij} = \infty$ for $j \notin \mathcal{T}_i$. We call $A = (a_{11}, a_{12}, \ldots, a_{cm_c})$ a *Mal'cev basis associated to the central series.*

For convenience, we will also use a simplified notation, in which the generators a_{ij} and exponents the e_{ij} are renumbered by replacing each subscript ij with $j + \sum\limits_{\ell<i} m_\ell$, so the generating sequence A can be written as $A = (a_1, \ldots, a_m)$. We allow the expression ij to stand for $j + \sum\limits_{\ell<i} m_\ell$ in other notations as well. We also denote $\mathcal{T} = \{i \mid e_i < \infty\}$. By the choice of $\{a_1, \ldots, a_m\}$, every element $g \in G$ may be written uniquely in the form $g = a_1^{\alpha_1} \cdots a_m^{\alpha_m}$, where $\alpha_i \in \mathbb{Z}$ and $0 \leq \alpha_i < e_i$ whenever $i \in \mathcal{T}$. The m-tuple $(\alpha_1, \ldots, \alpha_m)$ is called the *coordinate vector* or *Mal'cev coordinates* of g and it is denoted by $\operatorname{Coord}(g)$; the expression $a_1^{\alpha_1} \cdots a_m^{\alpha_m}$ is called the *(Mal'cev) normal form* of g. Moreover, we write $\operatorname{Coord}_i(g) = \alpha_i$.

To a Mal'cev basis A we associate a presentation of G as follows. For each $1 \leq i \leq m$, let n_i be such that $a_i \in G_{n_i} \setminus G_{n_i+1}$. If $i \in \mathcal{T}$, then $a_i^{e_i} \in G_{n_i+1}$, hence a relation

$$a_i^{e_i} = a_\ell^{\mu_{i\ell}} \cdots a_m^{\mu_{im}} \tag{5.1}$$

holds in G for $\mu_{ij} \in \mathbb{Z}$ and $\ell > i$ such that $a_\ell, \ldots, a_m \in G_{n_i+1}$. We call this the *power relation* for a_i. Moreover, for $1 \leq i < j \leq m$ we have relations of the form

$$a_j a_i = a_i a_j a_\ell^{\alpha_{ij\ell}} \cdots a_m^{\alpha_{ijm}} \qquad\qquad a_j^{-1} a_i = a_i a_j^{-1} a_\ell^{\beta_{ij\ell}} \cdots a_m^{\beta_{ijm}} \tag{5.2}$$

for $\alpha_{ijk}, \beta_{ijk} \in \mathbb{Z}$ and $l > j$ such that $a_\ell, \ldots, a_m \in G_{n_j+1}$.

A presentation with relations of the form (5.1)–(5.2) for all i resp. i and j is called a *nilpotent presentation.* Indeed, any presentation of this form will define a nilpotent group. It is called *consistent* if the order of a_i modulo $\langle a_{i+1}, \ldots, a_m \rangle$ is precisely e_i for all i. While presentations of this form need not, in general, be consistent, those derived from a central series of a group G as above are consistent. Given a consistent nilpotent presentation, there is an easy way to solve the word problem: simply apply the rules of the form (5.2) to move all occurrences of $a_1^{\pm 1}$ in the input word to the left, then apply the power relations (5.1) to reduce their number modulo e_1; finally, continue with a_2 and so on.

Multiplication functions. A crucial feature of the coordinate vectors for nilpotent groups is that the coordinates of a product $(a_1^{\alpha_1} \cdots a_m^{\alpha_m})(a_1^{\beta_1} \cdots a_m^{\beta_m})$ may be computed as a "nice" function (polynomial if $\mathcal{T} = \emptyset$) of the integers $\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_m$.

Lemma 5.1 ([Hal69, KM79])**.** *Let G be a nilpotent group with Mal'cev basis $a_1, \ldots, a_m$ and $\mathcal{T} = \emptyset$. There exist $p_1, \ldots, p_m \in \mathbb{Z}[x_1, \ldots, x_m, y_1, \ldots, y_m]$ and $q_1, \ldots, q_m \in \mathbb{Z}[x_1, \ldots, x_m, z]$ such that for $g, h \in G$ with* $\operatorname{Coord}(g) = (\gamma_1, \ldots, \gamma_m)$ *and* $\operatorname{Coord}(h) = (\delta_1, \ldots, \delta_m)$ *and $l \in \mathbb{Z}$ we have*

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

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

(iii) $\mathrm{Coord}_1(gh) = \gamma_1 + \delta_1$ *and* $\mathrm{Coord}_1(g^l) = l\gamma_1$.

All the proofs in this paper rely essentially on the fact that in a free nilpotent group multiplication of elements can be computed using these functions. For further background on nilpotent groups we refer to [Hal69, KM79].

Presentation of subgroups. Let $h_1, \ldots, h_n$ be elements of G given in normal form by $h_i = a_1^{\alpha_{i1}} \cdots a_m^{\alpha_{im}}$, for $i = 1, \ldots, n$, and let $H = \langle h_1, \ldots, h_n \rangle$. We associate the *matrix of coordinates*

$$A = \begin{pmatrix} \alpha_{11} & \cdots & \alpha_{1m} \\ \vdots & \ddots & \vdots \\ \alpha_{n1} & \cdots & \alpha_{nm} \end{pmatrix}$$

to the tuple $(h_1, \ldots, h_n)$ and conversely, to any $n \times m$ integer matrix, we associate an n-tuple of elements of G whose Mal'cev coordinates are given as the rows of the matrix and the subgroup H generated by the tuple. For each $i = 1, \ldots, n$ where row i is non-zero, let π_i be the column of the first non-zero entry ('pivot') in row i. The sequence $(h_1, \ldots, h_n)$ is said to be in *standard form* if the matrix of coordinates A is in row-echelon form and its pivot columns are maximally reduced, more specifically, if A satisfies the following properties:

(i) all rows of A are non-zero (i.e. no h_i is trivial),

(ii) $\pi_1 < \pi_2 < \cdots < \pi_s$ (where s is the number of pivots),

(iii) $\alpha_{i\pi_i} > 0$, for all $i = 1, \ldots, n$,

(iv) $0 \leq \alpha_{k\pi_i} < \alpha_{i\pi_i}$, for all $1 \leq k < i \leq s$

(v) if $\pi_i \in \mathcal{T}$, then $\alpha_{i\pi_i}$ divides e_{π_i}, for $i = 1, \ldots, s$.

The sequence (resp. matrix) is called *full* if in addition

(vi) $H \cap \langle a_i, a_{i+1}, \ldots, a_m \rangle$ is generated by $\{h_j \mid \pi_j \geq i\}$, for all $1 \leq i \leq m$.

Note that $\{h_j \mid \pi_j \geq i\}$ consists of those elements having 0 in their first $i - 1$ coordinates.

Quotient Mal'cev presentations. Let $c, r \in \mathbb{N}$ be fixed. The free nilpotent group $F_{c,r}$ of class c and rank r is defined as $F_{c,r} = \langle\, a_1, \ldots, a_r \mid [x_1, \ldots, x_{c+1}] = 1 \text{ for } x_1, \ldots, x_{c+1} \in F_{c,r} \,\rangle$, i. e., $F_{c,r}$ is the r-generated group only subject to the relations that weight $c + 1$ commutators are trivial. Throughout, we fix a Mal'cev basis $A = (a_1, \ldots, a_m)$ (which we call the *standard Mal'cev basis*) associated to the lower central series of $F_{c,r}$ such that the associated nilpotent presentation consists only of relations of the form (5.2), $\{a_1, \ldots, a_r\}$ generates $F_{c,r}$, and all other Mal'cev generators are iterated commutators of $a_1, \ldots, a_r$.

Denote by $\mathcal{N}_{c,r}$ the set of r-generated nilpotent groups of class at most c. Every group $G \in \mathcal{N}_{c,r}$ is a quotient of the free nilpotent group $F_{c,r}$, i. e., $G = F_{c,r}/N$ for some normal subgroup $N \leq F_{c,r}$. Assume that $T = (h_1, \ldots, h_s)$ is a full sequence generating N. Adding T to the set of relators of the free nilpotent group yields a new nilpotent presentation. This presentation will be called *quotient presentation* of G. For inputs of algorithms, we assume that a quotient presentation is always given as its matrix of coordinates in full form. Depending whether the entries of the matrix are encoded in unary or binary, we call the quotient presentation to be given in *unary* or *binary*.

In the following we always assume that a quotient presentation is part of the input, but c and r are fixed. Later, we will show how to compute quotient presentations from an arbitrary presentation.

5.3 Results

Words over the generators ±1 of $\mathbb{Z}$ correspond to unary representations of integers. As a generalization of binary encoded integers, we use words with binary exponents[31]. Remember that words with binary exponents constitute a special case of power words; indeed, by the same arguments as for Theorem 6.1 (see below) in all cases where we use words with binary exponents as inputs we could also allow power words without changing the complexity. Nevertheless, since the results in [4] are stated in terms of words with binary exponents we stick to this formalism.

Word problem and computation of Mal'cev coordinates. The first result in [4] is on the word problem and computation of Mal'cev coordinates.

Theorem 5.2. *Let $c, r \geq 1$ be fixed and let $(a_1, \ldots, a_m)$ be the standard Mal'cev basis of $F_{c,r}$. The following problem is TC^0-complete: on input of $G \in \mathcal{N}_{c,r}$ given as a binary encoded quotient presentation and a word with binary exponents $w = w_1^{x_1} \cdots w_n^{x_n}$, compute integers $y_1, \ldots, y_m$ (in binary) such that $w = a_1^{y_1} \cdots a_m^{y_m}$ in G and $0 \leq y_i < e_i$ for $i \in \mathcal{T}$. Moreover, if the input is given in unary (both G and w), then the output is in unary.*

Note that the statement for unary inputs is essentially the one of [Rob93]. Be aware that in the formulation of the theorem, $\mathcal{T}$ and e_i for $i \in \mathcal{T}$ depend on the input group G. These parameters can be read from the full matrix of coordinates representing G. As an immediate consequence of Theorem 5.2, we obtain:

Corollary 5.3. *Let $c, r \geq 1$ be fixed. The uniform, binary version of the word problem for groups in $\mathcal{N}_{c,r}$ is TC^0-complete (where the input is given as in Theorem 5.2).*

The proof of Theorem 5.2 follows the outline given in Section 5.2; however, we cannot apply the rules (5.1)–(5.2) one by one. Instead we do only two steps for each generator: first apply all possible rules (5.2) in one step and then apply the rules (5.1) in one step.

Greatest common divisors. The problem of computing greatest common divisors (gcds) is an essential step for solving the subgroup membership problem. Indeed, consider the nilpotent group $\mathbb{Z}$ and let $a, b, c \in \mathbb{Z}$. Then $c \in \langle a, b \rangle$ if and only if $\gcd(a, b) \mid c$.

The *extended gcd problem* (ExtGCD) is the following problem: on input of *binary* encoded numbers $a_1, \ldots, a_n \in \mathbb{Z}$, compute $x_1, \ldots, x_n \in \mathbb{Z}$ such that $x_1a_1 + \cdots + x_na_n = \gcd(a_1, \ldots, a_n)$. Clearly this can be done in P using the Euclidean algorithm, but it is not known whether it is actually in NC. Indeed the currently best known parallel algorithm for ExtGCD runs in time $\mathcal{O}(n/\log n)$ [Sed17].

On the other hand, computing the gcd of numbers encoded in *unary* is straightforward in TC^0 by an exhaustive search. We know that there are $x_1, \ldots, x_n \leq |\max\{|a_1|, \ldots, |a_n|\}|$ with $x_1a_1 + \cdots + x_na_n = \gcd(a_1, \ldots, a_n)$. However, for computing these x_i, one cannot check all possible combinations of values in TC^0 because there are exponentially many. Still we succeeded to show the following:[32]

Proposition 5.4. *The following problem is in TC^0: Given integers $a_1, \ldots, a_n$ in unary, compute $x_1, \ldots, x_n \in \mathbb{Z}$ (either in unary or binary) such that $x_1a_1 + \cdots + x_na_n = \gcd(a_1, \ldots, a_n)$ and $|x_i| \leq (n+1)\left(\max\{|a_1|, \ldots, |a_n|\}\right)^2$.*

[31] Recall: a word with binary exponents is of the form $w_1^{x_1} \cdots w_n^{x_n}$ where the w_i are from a fixed generating set of the group and the x_i are binary encoded integers.

[32] Meanwhile the bounds in this result have been improved by Gretchen Ostheimer (private communication).

Matrix reduction and subgroup membership problem. The central result of the paper is the following. Its proof uses Proposition 5.4.

Theorem 5.5. *Let $c, r \in \mathbb{N}$ be fixed. The following problem is in TC^0: given a unary encoded quotient presentation of $G \in \mathcal{N}_{c,r}$ and $h_1, \dots, h_n \in G$, compute the full form of the associated matrix of coordinates encoded in unary and hence the unique full-form sequence $(g_1, \dots, g_s)$ generating $\langle h_1, \dots, h_n \rangle$. Moreover, if G and $h_1, \dots, h_n$ are given in binary, then the full-form sequence with binary coefficients can be computed in $\mathsf{TC}^0(\textsc{ExtGCD})$.*

There is a rather long list of applications of Theorem 5.5.

Corollary 5.6. *Let $c, r \in \mathbb{N}$ be fixed. The following problem is in TC^0 (resp. $\mathsf{TC}^0(\textsc{ExtGCD})$ for binary inputs): given a quotient presentation of $G \in \mathcal{N}_{c,r}$, elements $h_1, \dots, h_n \in G$ and $h \in G$, decide whether or not h is an element of the subgroup $H = \langle h_1, \dots, h_n \rangle$.*

Moreover, if $h \in H$, the circuit computes the unique expression $h = g_1^{\gamma_1} \cdots g_s^{\gamma_s}$ where $(g_1, \dots, g_s)$ is the full-form sequence for H with the γ_i encoded in unary (resp. binary).

Alternatively, for unary inputs, the output can be given as word $h = h_{i_1}^{\epsilon_1} \cdots h_{i_t}^{\epsilon_t}$ where $i_j \in \{1, \dots, n\}$ and $\epsilon_j = \pm 1$.

Corollary 5.7. *Let $c, r \in \mathbb{N}$ be fixed. The following is in TC^0 for unary inputs and in $\mathsf{TC}^0(\textsc{ExtGCD})$ for binary inputs:*

Input: *a quotient presentation for $G \in \mathcal{N}_{c,r}$ and elements $h_1, \dots, h_n \in G$.*

Output: *a consistent nilpotent presentation for $H = \langle h_1, \dots, h_n \rangle$ given by a list of generators $(g_1, \dots, g_s)$ and numbers $\mu_{ij}, \alpha_{ijk}, \beta_{ijk} \in \mathbb{Z}$ encoded in unary (resp. binary) for $1 \leq i < j < k \leq s$ representing the relations* (5.1)–(5.2)*.*

Be aware that the output in Corollary 5.7 is *not* a quotient presentation. Still, we have:

Proposition 5.8. *Let c and r be fixed integers. The following is in TC^0: given an arbitrary finite presentation with generators $a_1, \dots, a_r$ of a group $G \in \mathcal{N}_{c,r}$ (as a list of relators given as words over $\{a_1, \dots, a_r\}^{\pm 1}$), compute a quotient presentation of G (encoded in unary) and an explicit isomorphism. Moreover, if the relators are given as words with binary exponents, then the binary encoded quotient presentation can be computed in $\mathsf{TC}^0(\textsc{ExtGCD})$.*

The next two theorems are applications of Theorem 5.5. Their proofs follow essentially the proofs of their counterparts Theorems 4.1 and 4.6 of [MMNV15].

Theorem 5.9 (Kernels and preimages)**.** *Let $c, r \in \mathbb{N}$ be fixed. The following is in TC^0 for unary inputs and in $\mathsf{TC}^0(\textsc{ExtGCD})$ for binary inputs: on input of*

- *$G, H \in \mathcal{N}_{c,r}$ given as quotient presentations,*
- *a subgroup $K = \langle g_1, \dots, g_n \rangle \leq G$,*
- *a list of elements $h_1, \dots, h_n$ defining a homomorphism $\varphi : K \to H$ via $\varphi(g_i) = h_i$, and*
- *optionally, an element $h \in H$ guaranteed to be in the image of φ,*

compute a generating set X for the kernel of φ, and an element $g \in G$ such that $\varphi(g) = h$.

In case of unary inputs, X and g will be returned as words, and for binary inputs, as words with binary exponents.

Theorem 5.10 (Conjugacy Problem)**.** *Let $c, r \in \mathbb{N}$ be fixed. The following is in TC^0 for unary inputs and in $\mathsf{TC}^0(\textsc{ExtGCD})$ for binary inputs: On input of some $G \in \mathcal{N}_{c,r}$ given as quotient presentation and elements $g, h \in G$, either produce some $u \in G$ such that $g = u^{-1}hu$, or determine that no such element u exists. In case of unary inputs, u will be returned as a word, for binary inputs, as a word with binary exponents.*

5.4 Contribution by the author

The results summarized in this chapter have been published in the joint paper with Alexei Miasnikov [4]. It is based on previous work [MMNV15] by Alexei Miasnikov together with Jeremy Macdonald, Andrey Nikolaev, and Svetla Vassileva. Alexei Miasnikov provided the idea to start this work and helped significantly by giving advice and discussing the results. Apart from that, most of the extensions over [MMNV15] as well as the writing of the paper are due to the author.

6 The power word problem

6.1 Background

In this paper [2], the *power word problem* is introduced and studied. Recall that on input of a power word $(p_1, x_1, p_2, x_2, \ldots, p_n, x_n)$, where the x_i are integers encoded in binary, it asks whether $p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n} =_G 1$. As already observed, in terms of complexity, the power word problem lies somewhere in between the (ordinary) word problem and the compressed word problem.

Related work. Implicitly, variants of the power word problem have been studied before. In the commutative setting, Ge [Ge93] has shown that there is a polynomial time algorithm for verifying an identity $\alpha_1^{x_1} \alpha_2^{x_2} \cdots \alpha_n^{x_n} = 1$, where the α_i are elements of an algebraic number field and the x_i are binary encoded integers. Moreover, in [GS07], Gurevich and Schupp present a polynomial time algorithm for a compressed form of the subgroup membership problem for a free group F, where group elements are represented in the form $a_1^{x_1} a_2^{x_2} \cdots a_n^{x_n}$ with binary encoded integers x_i and the a_i are standard generators of the free group F. This is the same input representation as in [4] (Chapter 5) and is more restrictive then power words.

Another problem related to the power word problem is the knapsack problem [GKLZ18, LZ18, MNU15] for a finitely generated group G (with generating set Σ): given a sequence of words $w, w_1, \ldots, w_n \in \Sigma^*$, the question is whether there exist $x_1, \ldots, x_n \in \mathbb{N}$ such that $w =_G w_1^{x_1} \cdots w_n^{x_n}$. For many groups G one can show that if such $x_1, \ldots, x_n \in \mathbb{N}$ exist, then there exist such numbers of size $2^{\mathrm{poly}(N)}$, where $N = |w| + |w_1| + \cdots + |w_n|$ is the input length. For instance, this holds for right-angled Artin groups (also known as graph groups), which can be used to show that the knapsack problem in right-angled Artin groups belongs to NP.

In the very recent work [FGLZ20], Figelius, Ganardi, Lohrey and Zetzsche further investigate the power word problem and its relation to the knapsack problem. In particular, they establish that the power word problem for iterated wreath products of free abelian groups and for groups $G \wr \mathbb{Z}$ with G f.g. nilpotent is in TC^0 (thus, extending Theorem 6.5) and that, if G is uniformly SENS (for a definition, see Chapter 7), then the power word problem for $G \wr \mathbb{Z}$ is coNP-hard (extending the hardness part of Theorem 6.6). Applying these results to the knapsack problem yields NP-completeness for iterated wreath products of free abelian groups and Σ_p^2-hardness for $G \wr \mathbb{Z}$ if G is uniformly SENS.

6.2 Results

This paper considers the power word problem in several classes of groups: nilpotent groups, (virtually) free groups, the Grigorchuk group and certain wreath products. The results are summarized in Table 2. Here also the relation to the complexity of the ordinary word problem and of the compressed word problem is shown. In particular, we can see examples of groups G where $\textsc{PowerWP}(G)$ is indeed more difficult than $\mathrm{WP}(G)$ (under standard assumptions from complexity theory), as well as examples of groups G where $\textsc{PowerWP}(G)$ and $\mathrm{WP}(G)$ are equally difficult.

class of groups	PowerWP	CompressedWP	WP
nilpotent groups	TC^0	DET, $\mathsf{C_{=}L}$-hard [KL18a]	TC^0 [Rob93]
Grigorchuk group G	$\mathsf{LOGSPACE}^{a)}$	PSPACE-complete [1]	$\mathsf{LOGSPACE}$ [GZ91]
non-abelian f.g. free	$\mathsf{LOGSPACE}^{b)}$	P-complete [Loh06]	$\mathsf{LOGSPACE}$ [LZ77]
$G \wr \mathbb{Z}$ for G f.g. abelian	TC^0	coRP [KL18a]	TC^0 [3]
$G \wr \mathbb{Z}$ for G finite non-solvable	coNP-complete	PSPACE-complete [1]	NC^1 [Waa90]
$F_2 \wr \mathbb{Z}$	coNP-complete	PSPACE-complete [1]	$\mathsf{LOGSPACE}^{b)}$[Waa90]
finite extension of a f.g. group H	NC^1-many-one-reducible to PowerWP(H) (resp. CompressedWP(H) [KL18a], resp. WP(H) [Waa90])		

$^{a)}\mathsf{AC}^0$-many-one-reducible to WP(G).
$^{b)}\mathsf{AC}^0$-Turing-reducible to WP(F_2).

Table 2: Summary of results on the power word problem compared to results on the (compressed) word problem.

The first theorem (on nilpotent groups) is a rather easy consequence of the results of [4]. In order to show the connection to [4], we present its proof here.

Theorem 6.1. *If G is a finitely generated nilpotent group, then PowerWP(G) is in TC^0.*

Proof. Fix a quotient Mal'cev presentation for G as defined in Chapter 5. Given a power word $w = p_1^{x_1} p_2^{x_2} \cdots p_n^{x_n}$ as input, we first compute the Mal'cev coordinates of each p_i for $i = 1, \dots, n$ in parallel in TC^0 using Theorem 5.2. As a next step, we can use the power polynomials from Lemma 5.1 in order to compute the Mal'cev coordinates of $p_i^{x_i}$ for $i = 1, \dots, n$ in parallel. This also can be done in TC^0 since multiplication of integers is in TC^0. As a last step, we again apply the TC^0 circuit from Theorem 5.2 in order to compute the Mal'cev coordinates for the whole power word. Now $w =_G 1$ if and only if all the coordinates are zero. □

The main result on the power word problem essentially establishes that in free groups it is not more difficult than the ordinary word problem:

Theorem 6.2. *The power word problem for a finitely generated free group is AC^0-Turing-reducible to the word problem for the free group F_2.*

It is easy to see that the power word problem for every finite group belongs to NC^1. The following result generalizes this fact:

Theorem 6.3. *Let G be finitely generated and let $H \leq G$ have finite index. Then PowerWP(G) is NC^1-many-one-reducible to PowerWP(H).*

As an immediate consequence of Theorem 6.2, Theorem 6.3 and the NC^1-hardness of the word problem for F_2 [Rob93, Theorem 6.3] we obtain:

Corollary 6.4. *The power word problem for every finitely generated virtually free group is AC^0-Turing-reducible to the word problem for the free group F_2.*

Theorem 6.5. *For every finitely generated abelian group G, PowerWP$(G \wr \mathbb{Z})$ is in TC^0.*

Theorem 6.6. *Let G be either a finite non-solvable group or a finitely generated free group of rank at least two. Then PowerWP$(G \wr \mathbb{Z})$ is coNP-complete.*

Theorem 6.7. *The power word problem for the Grigorchuk group is* AC^0*-many-one-reducible to its word problem.*

It is well-known that the word problem for the Grigorchuk group is in $\mathsf{LOGSPACE}$ (see e. g. [GZ91]). Thus, also the power word problem is in $\mathsf{LOGSPACE}$. The proof of Theorem 6.7 relies on [BGv03, Theorem 6.6], which shows that in the Grigorchuk group every element of length N has order at most $CN^{3/2}$ for some constant C.

6.3 Outline of the proof of Theorem 6.2

The proof of Theorem 6.2 consists of two main steps: First, we do some preprocessing leading to a particularly nice instance of the power word problem. While this preprocessing is simple from a theoretical point of view, it is where the main part of the workload is performed during the execution of the algorithm. Then, in the second step, all exponents are reduced to polynomial size. After this shortening process, the power word problem can be solved by the algorithm for the ordinary word problem. The most involved part is to prove correctness of the shortening process.

Preprocessing. Fix an arbitrary order on the input alphabet Σ. This gives us the lexicographic order on Σ^*, which is denoted by $\preceq$. Let $\Omega \subseteq \Sigma^*$ denote the set of words w such that

- w is non-empty,
- w is cyclically reduced (i.e, w does not contain a factor aa^{-1} for $a \in \Sigma$ and cannot be written as aua^{-1} for $a \in \Sigma$),
- w is primitive (i.e, w cannot be written as u^n for $n \geq 2$),
- w is lexicographically minimal among all cyclic permutations of w and w^{-1} (i.e., $w \preceq uv$ for all $u, v \in \Sigma^*$ with $vu = w$ or $vu = w^{-1}$).

Notice that Ω consists of Lyndon words [Lot83, Chapter 5.1] with the stronger requirement of being freely reduced, cyclically reduced and also minimal among the conjugacy class of the inverse. The first aim is to rewrite the input power word in the form

$$w = s_0 p_1^{x_1} s_1 \cdots p_n^{x_n} s_n \qquad \text{with } p_i \in \Omega \text{ and } s_i \in \mathrm{IRR}(S). \tag{6.1}$$

The reason for this lies in the following crucial lemma which essentially says that, if a long factor of $p_i^{x_i}$ cancels with some $p_j^{x_j}$, then already $p_i = p_j$. Thus, only the same p_i can cancel implying that we can make the exponents of the different p_i independently smaller. This is made precise in the following lemma:

Lemma 6.8. *Let* $p, q \in \Omega$*,* $x, y \in \mathbb{Z}$ *and let* v *be a factor of* p^x *and* w *a factor of* q^y*. If* $vw =_{F_X} 1$ *and* $|v| = |w| \geq |p| + |q| - 1$*, then* $p = q$*.*

Lemma 6.9. *The following is in* $\mathsf{AC}^0(\mathrm{WP}(F_2))$*: given a power word* v*, compute a power word* w *of the form* (6.1) *such that* $v =_{F_X} w$*.*

The shortening process. The idea of the shortening process is that we replace each large power $p_i^{x_i}$ by some $p_i^{y_i}$ with y_i polynomial in the input length. In order to prove correctness of the shortening process (i. e., that $w =_{F_X} 1$ if and only if the shortened version of w is 1 in F_X), we need the uniqueness conditions established during the preprocessing phase. The proof relies on some rewriting techniques.

Putting things together. Once we have established the correctness of the shortening process, the proof of Theorem 6.2 is straightforward: We start with the preprocessing as described in Lemma 6.9 leading to a word $w = s_0 p_1^{x_1} s_1 \cdots p_n^{x_n} s_n$ with $p_i \in \Omega$ and $s_i \in \mathrm{IRR}(S)$ as in (6.1). After that we apply the shortening procedure for all $p \in \{p_i \mid 1 \leq i \leq n\}$. This can be done in parallel for all p, as the outcome of the shortening only depends on the p-exponents. This leads to a word $\hat{w}$ of polynomial length. Finally, we can test whether $\hat{w} =_{F_X} 1$ using one oracle gate for $\mathrm{WP}(F_2)$ (recall that F_2 contains a copy of F_X). The computations for shortening only involve iterated addition (and comparisons of integers), which is in TC^0 and, thus, can be solved in AC^0 with oracle gates for $\mathrm{WP}(F_2)$.

6.4 Contribution by the author

This chapter summarizes [2], which is a joint publication of Markus Lohrey and the author. The main contribution by the author is the proof of the main result Theorem 6.2. Theorem 6.5 has been discovered by Markus Lohrey. The other results were obtained in close collaboration.

7 Groups with ALOGTIME-hard word problems and PSPACE-complete compressed word problems.

7.1 Background

A striking connection between the word problem for groups and complexity theory was established by Barrington [Bar89]: for every finite non-solvable group G, the word problem of G is complete for ALOGTIME, which is the same as DLOGTIME-uniform NC^1. Moreover, the reduction is as simple as it could be: every output bit depends on only one input bit. Thus, one can say that ALOGTIME is completely characterized via group theory. Moreover, this idea has been extended to characterize ACC^0 by solvable monoids [BT88]. On the other hand, the word problem of a finite p-group is in $\mathsf{ACC}^0[p]$, so Smolensky's lower bound [Smo87] implies that it is strictly easier than the word problem of a finite non-solvable group.

Barrington's construction is based on the observation that an AND-gate can be simulated by a commutator. This explains the connection to non-solvability. In this light, it seems natural that the word problem of finite p-groups is not ALOGTIME-hard: they are all nilpotent, so iterated commutators eventually become trivial. For infinite groups, a construction similar to Barrington's was used by Robinson [Rob93] to show that the word problem of a non-abelian free group is ALOGTIME-hard. Since by [LZ77] the word problem of a free group is in LOGSPACE, the complexity is narrowed down quite precisely (although no completeness is known).

Compressed word problems. In the second part of the paper we study the *compressed word problem* for a finitely generated group G, $\textsc{CompressedWP}(G)$ for short. Recall that the compressed word problem for a group G receives as input an SLP producing a string w in Σ^* and the question is whether $w = 1$ in G.

Compressed word problems for finite groups have been studied in [BMPT97]: $\textsc{CompressedWP}(G)$ is P-complete for finite non-solvable groups (by a Barrington style argument) and in NC^2 for finite solvable groups. The compressed word problem for linear groups is tightly related to PIT (polynomial identity testing, i.e., the question whether a circuit over a polynomial ring evaluates to the zero-polynomial; see e.g. [SY10]): For every f.g. linear group the compressed word problem reduces in polynomial time to PIT for $\mathbb{Z}[x]$ or $\mathbb{F}[x]$ and hence belongs to coRP, the complement of randomized polynomial time [Loh14, Theorem 4.15]. Moreover, the compressed word problem for the group $\mathrm{SL}_3(\mathbb{Z})$ is equivalent to PIT for $\mathbb{Z}[x]$ with respect to polynomial time reductions [Loh14, Theorem 4.16].

Here, we are interested in the compressed word problem for wreath products of the form $G \wr \mathbb{Z}$ for finitely generated groups G. König and Lohrey gave a coRP algorithm for $\textsc{CompressedWP}(G \wr \mathbb{Z})$ for abelian groups G using a reduction to PIT. On the other hand, if G is non-abelian, $\textsc{CompressedWP}(G \wr \mathbb{Z})$ is coNP-hard [Loh14].

7.2 Results

Strongly efficiently non-solvable groups and ALOGTIME. The first contribution of this paper is to identify the essence of Barrington's and Robinson's constructions. For this we introduce a strengthened condition of non-solvability.

Definition 7.1. We call a group G with the finite standard generating set Σ *uniformly strongly efficiently non-solvable (uniformly SENS)* if there is a constant $\mu \in \mathbb{N}$ and words $g_{d,v} \in \Sigma^*$ for all $d \in \mathbb{N}$, $v \in \{0,1\}^{\leq d}$ such that

(a) $|g_{d,v}| = 2^{\mu d}$ for all $v \in \{0,1\}^d$,

(b) $g_{d,v} = [g_{d,v0}, g_{d,v1}]$ for all $v \in \{0,1\}^{<d}$ (here we take the commutator of words),

(c) $g_{d,\varepsilon} \neq 1$ in G, and

(d) given $v \in \{0,1\}^d$, a positive integer i encoded in binary with μd bits, and $a \in \Sigma$ one can decide in $\mathsf{DLINTIME}$ whether the i-th letter of $g_{d,v}$ is a.

If G is required to only satisfy (a)–(c), then G is called SENS.

In a SENS group G, non-solvability is witnessed by efficiently computable balanced nested commutators of arbitrary depth that are non-trivial in G. Moreover, SENS groups enjoy the following properties:

- The property of being (uniformly) SENS is independent of the choice of the standard generating set.
- If $Q = H/K$ is a f.g. subquotient (quotient of a subgroup) of a f.g. group G and Q is (uniformly) SENS, then G is also (uniformly) SENS.
- If G is (uniformly) SENS, then $G/Z(G)$ is (uniformly) SENS.
- If G is a finite non-solvable group, then G is uniformly SENS. (For $G = A_5$, this is the heart of Barrington's argument.) Thus, for finite groups, being uniformly SENS is equivalent to being non-solvable.

In particular, every group having a finite non-solvable subquotient is uniformly SENS. Since every free group of rank $n \geq 2$ projects to a finite non-solvable group, we get:

Corollary 7.2. *A f.g. free group of rank $n \geq 2$ is uniformly SENS.*

By following Barrington's arguments we show:

Theorem 7.3. *Let G be uniformly SENS. Then* $\mathrm{WP}(G)$ *is hard for* $\mathsf{ALOGTIME}$ *under* $\mathsf{DLOGTIME}$*-reductions (and also* $\mathsf{DLOGTIME}$*-uniform projection reductions or* AC^0*-reductions).*

That means that for every non-solvable group G, the word problem for G is $\mathsf{ALOGTIME}$-hard, unless the word length of the G-elements witnessing the non-solvability grows very fast (in the preprint [BFLW19] we give an example of a non-solvable group where the latter happens) or these elements cannot be computed efficiently.

Examples of SENS groups. In order to find examples of SENS groups without finite non-solvable subquotients, we present in the [BFLW19] a general criterion that implies the uniform SENS-condition. Using this general criterion, we show:

Theorem 7.4. *Let G be a finitely generated group with $G \wr H \leq G$ for some non-trivial group H. Then G is uniformly SENS.*

Theorem 7.5. *Let G be a weakly branched self-similar group, and assume that it admits a f.g. branching subgroup K. Then K and hence G are uniformly SENS.*

As an application of Theorem 7.4 and Theorem 7.5, we can show ALOGTIME-hardness of the word problems for several famous groups:

Corollary 7.6. *The word problems for the following groups are hard for* ALOGTIME*:*

- *Thompson's groups,*
- *weakly branched self-similar groups with a finitely generated branching subgroup (in particular, Grigorchuk's group and the Gupta-Sidki groups).*

Contracting self-similar groups. Recall the notation $g@x$ for the coordinates of $\varphi(g)$. We iteratively define $g@v = g@x_1 \cdots @x_n$ for any word $v = x_1 \cdots x_n \in X^*$. A self-similar group G is called *contracting* if there is a finite subset $N \subseteq G$ such that, for all $g \in G$, we have $g@v \in N$ whenever v is long enough (depending on g), see also [Nek05, Definition 2.11.1].

It is well-known that the Grigorchuk group and the Gupta-Sidki groups are contracting. The following result has been quoted numerous times, but has never appeared in print (although the algorithm has been implicitly described in [Nek05]). Therefore, a proof is given in the arXiv version of the summarized paper. A proof for the Grigorchuk group may be found in [GZ91]:

Proposition 7.7. *Let G be a f.g. contracting self-similar group. Then* $\mathrm{WP}(G)$ *can be solved in* LOGSPACE*.*

In particular, together with Corollary 7.6 we obtain a quite precise description of the complexity of the word problems of the Grigorchuk group and the Gupta-Sidki groups: they are in LOGSPACE and ALOGTIME-hard – thus, we have the same knowledge like for the complexity of the word problem of a (non-abelian) free group. Since membership results for ALOGTIME as well as LOGSPACE-hardness proofs seem to be out of reach, one future point of research might address the question whether the word problem of the Grigorchuk group reduces to the word problem of the free group or vice-versa or whether neither of the two possibilities holds.

A dichotomy for linear groups. In [KL18a, Theorem 7], König and Lohrey showed that the word problem of every f.g. solvable linear group is in TC^0. They asked the question whether there is a dichotomy in the sense that the word problem of a linear group either is in TC^0 or ALOGTIME-hard. As another application of the SENS condition, we can answer this question affirmatively using the famous Tits' alternative [Tit72]:

Corollary 7.8. *For every f.g. linear group the word problem either is in* DLOGTIME*-uniform* TC^0 *or the word problem is* ALOGTIME*-hard.*

Compressed word problems for wreath products. Our main result for the compressed word problem in wreath products pinpoints the exact complexity of $\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ for the case that G has a trivial center. Theorem 7.9 below uses the concept of leaf languages [BCS92, Her97, HLS+93, HVW96, JMT96]:

For a language $K \subseteq \Gamma^*$ over a finite alphabet Γ consider nondeterministic polynomial time machines M that after termination print a symbol from Γ on every computation path. Moreover, fix a linear ordering on the transition tuples of M. As a consequence the computation tree $T(x)$ for a machine input x becomes a finite ordered tree. The corresponding leaf string $\mathrm{leaf}(M, x)$ is obtained by listing symbols from Γ that are printed in the leafs of $T(x)$ from left to right. The class $\mathsf{LEAF}(K)$ consists of all languages L for which there exists a nondeterministic polynomial time machine as described above such that $x \in L$ if and only if $\mathrm{leaf}(M, x) \in K$. As a prototypical example we have $\mathsf{NP} = \mathsf{LEAF}(\{0,1\}^*1\{0,1\}^*)$. Here, we are interested in leaf language classes where K is the word problem for a f.g. group. It is easy to see that the generating set for the group has no influence on $\mathsf{LEAF}(\mathrm{WP}(G))$.

For a complexity class C we denote by $\forall\mathsf{C}$ the class of all languages L such that there exists a polynomial $p(n)$ and a language $K \in \mathsf{C}$ with $L = \{u \mid \forall v \in \{0,1\}^{p(|u|)} : u\#v \in K\}$ (e. g. $\forall\mathsf{P} = \mathsf{coNP}$ and $\forall\mathsf{PSPACE} = \mathsf{PSPACE}$). Moreover, for a complexity class C the class $\mathsf{MOD}_m\mathsf{C}$ is defined by $L \in \mathsf{MOD}_m\mathsf{C}$ if there exists a polynomial $p(n)$ and a language $K \in \mathsf{C}$ such that $L = \{u \mid |\{v \in \{0,1\}^{p(|u|)} : u\#v \in K\}| \not\equiv 0 \mod m\}$.

Theorem 7.9. *Let G be a f.g. non-trivial group with center $Z = Z(G)$.*

- *$\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ belongs to $\forall\mathsf{LEAF}(\mathrm{WP}(G))$.*
- *$\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ is hard for the class $\forall\mathsf{LEAF}(\mathrm{WP}(G/Z))$.*

In particular, if $Z = 1$, then $\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ is complete for $\forall\mathsf{LEAF}(\mathrm{WP}(G))$.

Example 7.10. If G is a finite non-abelian p-group, then $\mathsf{LEAF}(\mathrm{WP}(G)) \subseteq \mathsf{MOD}_p \cdots \mathsf{MOD}_p\mathsf{P} = \mathsf{MOD}_p\mathsf{P} \subseteq \mathsf{LEAF}(\mathrm{WP}(G))$ by [Her94, Satz 4.32], [BG92, Theorem 6.7], and [Her00, Theorem 2.2] and likewise $\mathsf{LEAF}(\mathrm{WP}(G/Z(G))) = \mathsf{MOD}_p\mathsf{P}$. Hence, in this case $\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ is complete for $\forall\mathsf{MOD}_p\mathsf{P}$.

Example 7.11. Consider the symmetric group on three elements S_3. By [Her00, Example 2.5] we have $\mathsf{LEAF}(\mathrm{WP}(S_3)) = \mathsf{MOD}_3\mathsf{MOD}_2\mathsf{P}$ (also written as $\mathsf{MOD}_3{\oplus}\mathsf{P}$). Since S_3 has trivial center, it follows that $\textsc{Compressed}\mathrm{WP}(S_3 \wr \mathbb{Z})$ is complete for $\forall\mathsf{MOD}_3{\oplus}\mathsf{P}$.

Corollary 7.12. *If G is uniformly SENS, then $\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ is PSPACE-hard.*

Corollary 7.13. *The compressed word problem for the following groups is PSPACE-complete:*

(i) *wreath products $G \wr \mathbb{Z}$ where G is finite non-solvable or free of rank at least two,*

(ii) *Thompson's groups,*

(iii) *the Grigorchuk group, and*

(iv) *all Gupta-Sidki groups.*

Thus, in particular, the word problem of the Grigorchuk group is in $\mathsf{LOGSPACE}$ (Proposition 7.7) while its compressed word problem is PSPACE-complete – hence, provably more difficult. Moreover, by Theorem 6.7 also the power word problem of the Grigorchuk group is in $\mathsf{LOGSPACE}$ – so, the compressed word problem is even provably more difficult than the power word problem. Notice that previously Wächter and the author constructed an automaton group with a PSPACE-complete word problem and an $\mathsf{EXPSPACE}$-complete compressed word problem [WW20] by encoding a Turing machine into the group.

In order to derive Corollary 7.13 from Theorem 7.9, we use a padded version of Theorem 7.3 saying that PSPACE is contained in $\mathsf{LEAF}(\mathrm{WP}(G/Z(G)))$ (this yields PSPACE-hardness of $\textsc{Compressed}\mathrm{WP}(G \wr \mathbb{Z})$ for every SENS group G). For Thompson's groups, the Grigorchuk group, and the Gupta-Sidki groups we also use a certain self-embedding property: for all these groups G a wreath product $G \wr A$ embeds into G for some $A \neq 1$. Thompson's group F has this property for $A = \mathbb{Z}$ [GS99]. For the Grigorchuk group and the Gupta-Sidki groups we show that one can take $A = \mathbb{Z}/p$ for some $p \geq 2$.

7.3 Contribution by the author

This chapter summarizes the conference paper [1]. The results have been obtained together with Laurent Bartholdi, Micheal Figelius, and Markus Lohrey.

Besides initiating the study of the complexity of the word problem of the Grigorchuk group, the main contribution by the author is in the first part of the paper, namely the definition of SENS groups, basic properties of SENS groups and the $\mathsf{ALOGTIME}$-hardness proof for the word problem of SENS groups. The author also contributed ideas towards the PSPACE-hardness proof of the compressed word problem of the Grigorchuk group as well as some minor details like Examples 7.10 and 7.11.

8 Hardness of equations over finite solvable groups under the exponential time hypothesis.

8.1 Background

This chapter summarizes the results on equations in finite solvable groups obtained in [6]. The study of equations over algebraic structures has a long history in mathematics. Some of the first explicit decidability results in group theory are due to Makanin [Mak84], who showed that equations over free groups are decidable. Subsequently, several other decidability and undecidability results as well as complexity results on equations over infinite groups emerged (see [DE17, GMO20, LS06, Rom79] for a random selection). For a fixed group G, the equation satisfiability problem EQN-SAT is as follows: given an expression $\alpha \in (G \cup \mathcal{X} \cup \mathcal{X}^{-1})^*$ where $\mathcal{X}$ is some set of variables, the question is whether there exists some assignment $\sigma : \mathcal{X} \to G$ such that $\sigma(\alpha) = 1$. Likewise EQN-ID is the problem to decide on input of an expression whether it evaluates to 1 under *all* assignments.

Henceforth, all groups we consider are finite. In this case, equation satisfiability can be solved in NP by a simple guess and check algorithm. Still the complexity is an interesting topic of research: its study has been initiated by Goldmann and Russell [GR02] showing that satisfiability of systems of equations can be decided in P if and only if the group is abelian (assuming $\mathsf{P} \neq \mathsf{NP}$) – otherwise, the problem is NP-complete. They also obtained results for single equations: EQN-SAT is NP-complete for non-solvable groups, while for nilpotent groups it is in P. The case of solvable but non-nilpotent groups remained open. Indeed, Burris and Lawrence raised the question whether EQN-ID$(G) \in \mathsf{P}$ for all finite solvable groups G [BL04, Problem 1]. Moreover, Horváth [Hor11] conjectured a positive answer.

Related work on equations. Since the work of Goldman and Russell [GR02] and Barrington et. al. [BMM$^+$00], a long list of literature has appeared investigating EQN-ID and EQN-SAT in groups and other algebraic structures. In [BL04] it is shown that EQN-ID is in P for nilpotent groups as well as for dihedral groups D_k where k is odd. In [Hor15, HS06, Föl17] these results were extended to some further classes of groups. The latest result is due to Földvári and Horváth [FH20], who established that EQN-SAT is in P for the semidirect product of a p-group and an abelian group and that EQN-ID is in P for nilpotent-by-abelian groups. Notice that all these groups have in common that their Fitting length is at most two.

In [HS11, HS12] the EQN-SAT and EQN-ID problems for generalized terms are introduced. A generalized term is an expression which may also use commutators or even more complicated operations inside. Using commutators is a more succinct representation, which allows to show that EQN-SAT is NP-complete and EQN-ID is coNP-complete in the alternating group A_4 [HS12]. In [Kom19] this result is extended by showing that with extended terms EQN-SAT is NP-complete and EQN-ID is coNP-complete for all non-nilpotent groups.

Exponential time hypothesis. The exponential time hypothesis (ETH) is the conjecture that there is some $\delta > 0$ such that every algorithm for 3SAT needs time $\Omega(2^{\delta n})$ in the worst case (where n is the number of variables of the given 3SAT instance). By the sparsification lemma [IPZ01, Thm. 1] this is equivalent to the existence of some $\epsilon > 0$ such that every algorithm for 3SAT needs time $\Omega(2^{\epsilon(m+n)})$ in the worst case (where m is the number of clauses). In particular, under ETH there is no algorithm for 3SAT running in time $2^{o(n+m)}$.

Equations in groups. An *expression* (also called a *polynomial* in [SS06, HS06, Kom19]) over a group G is a word α over the alphabet $G \cup \mathcal{X} \cup \mathcal{X}^{-1}$ where $\mathcal{X}$ is a set of variables. Here $\mathcal{X}^{-1}$ denotes a formal set of inverses of the variables.

An assignment for an expression α is a mapping $\sigma : \mathcal{X} \to G$ (here σ is canonically extended by $\sigma(X^{-1}) = \sigma(X)^{-1}$ and $\sigma(g) = g$ for $g \in G$). An assignment σ is *satisfying* if $\sigma(\alpha) = 1$ in G. The problems EQN-SAT(G) and EQN-ID(G) are as follows: for both of them the input is an expression α. For EQN-SAT(G) the question is whether there *exists* a satisfying assignment, for EQN-ID(G) the question is whether *all* assignments are satisfying.

In the literature EQN-SAT is also denoted by POL-SAT [SS06, HS06] or Eq [Kom19], while EQN-ID is also referred to as POL-EQ (e. g. [SS06, HS06, KTT07]) or Id [Kom19].

Fitting length. The *Fitting* subgroup Fit(G) is the union of all nilpotent normal subgroups. Let G be a finite solvable group. It is well-known that Fit(G) itself is a nilpotent normal subgroup (see e. g. [Hup67, Satz 4.2]). The *upper Fitting series*

$$1 = \mathcal{U}_0 G \lhd \mathcal{U}_1 G \lhd \cdots \lhd \mathcal{U}_k G = G$$

is defined by $\mathcal{U}_{i+1}G/\mathcal{U}_i G = \mathrm{Fit}(G/\mathcal{U}_i G)$ and k is called the *Fitting length* of G. Notice that the Fitting length is the smallest d such that there is a sequence $1 = G_0 \trianglelefteq \cdots \trianglelefteq G_d = G$ with all quotients G_{i+1}/G_i nilpotent.

8.2 Results

In this work, assuming the exponential time hypothesis, we give a negative answer to Burris and Lawrence's question whether EQN-ID $\in$ P for all finite solvable groups. Thus, we derive the first lower bounds for EQN-ID and EQN-SAT in finite solvable groups:

Theorem 8.1. *Let G be a finite solvable group and assume that either*

- *the Fitting length of G is at least four, or*
- *the Fitting length of G is three and $G/\mathcal{U}_2 G$ is not a 2-group.*

Then EQN-SAT(G) and EQN-ID(G) are not in P *under the exponential time hypothesis.*

The case that $G/\mathcal{U}_2 G$ is a 2-group. In the recent paper [IKK20] Idziak, Kawałek, and Krzaczkowski proved a $2^{\Omega(\log^2(n))}$-lower bound under ETH for EQN-SAT(S_4) (where S_4 is the symmetric group on four elements; it has Fitting length three and $\mathcal{U}_2 S_4 = A_4$ has index two). A new joint paper [IKKW20] unifying our approaches and proving Theorem 8.1 for *all* groups of Fitting length 3 is has been completed after the submission of this Habilitationsschrift.

Equations in finite semigroups. For a semigroup S, the problem EQN-SAT(S) receives two expressions as input. The question is whether the two expressions evaluate to the same element under some assignment. For semigroups R, S we say that R *divides* S if R is a quotient of a subsemigroup of S.

Corollary 8.2. *Let S be a finite semigroup and G a group dividing S. If G meets the requirements of Theorem 8.1, then EQN-SAT(S) is not in* P *under ETH.*

G-programs and AND-weakness. Let G be a finite group. An n-input G-program of length ℓ with variables (input bits) from $\{B_1, \ldots, B_n\}$ is a sequence

$$P = \langle B_{i_1}, a_1, b_1\rangle\langle B_{i_2}, a_2, b_2\rangle \cdots \langle B_{i_\ell}, a_\ell, b_\ell\rangle \in (\{B_1, \ldots, B_n\} \times G \times G)^*.$$

For an assignment $\sigma : \{B_1, \ldots, B_n\} \to \{0,1\}$ we define $\sigma(P) \in G$ as the group element $c_1c_2 \cdots c_\ell$, where $c_j = a_j$ if $B_{i_j} = 0$ and $c_j = b_j$ if $B_{i_j} = 1$ for all $1 \leq j \leq \ell$. We say that an n-input G-program P *computes* a function $f : \{0,1\}^n \to \{0,1\}$ if P is over the variables $B_1, \ldots, B_n$ and there is some $S \subseteq G$ such that $\sigma(P) \in S$ if and only if $f(\sigma) = 1$.

Consequences for ProgramSAT. ProgramSAT is the following problem: given a G-program P with variables $B_1, \ldots, B_n$, decide whether there is an assignment $\sigma : \{B_1, \ldots, B_n\} \to G$ such that $\sigma(P) = 1$. Since EQN-SAT(G) $\leq_{\mathrm{m}}^{\mathsf{AC}^0}$ ProgramSAT(G) for finite groups G by [BMM$^+$00, Lem. 1], ProgramSAT(G) is not in P under ETH if G meets the conditions from Theorem 8.1.

The AND-weakness conjecture. In [BST90], Barrington, Straubing and Thérien conjectured that, if G is finite and solvable, every G-program computing the n-input AND requires length exponential in n. This is called the AND-*weakness conjecture*. There are various interpretations of the term "exponential" in literature: while often it means $2^{\Omega(n)}$, in other occasions it is used for $2^{n^{\Omega(1)}}$. We also give a new proof that when interpreting "exponential" as $2^{\Omega(n)}$, the AND-weakness conjecture does *not* hold (as remarked in the earlier paper [BBR94]).

Corollary 8.3. *Let G be a finite solvable group of Fitting length $d \geq 2$. Then the n-input* AND *function can be computed by a G-program of length $2^{\mathcal{O}(n^{1/(d-1)})}$.*

Conjecture 8.4 (AND-weakness [BST90])**.** Let G be finite solvable. Then every G-program for the n-input AND has length $2^{n^{\Omega(1)}}$.

Notice that [BMM$^+$00, Theorem 2] (if G is AND-weak, ProgramSAT over G can be decided in quasi-polynomial time) still holds with this version of the AND-weakness conjecture. Hence, Conjecture 8.4 implies the following conjecture:

Conjecture 8.5. If G is a finite solvable group, then EQN-SAT(G) and EQN-ID(G) are decidable in quasipolynomial time.

Also notice that for arbitrary finitely generated groups, the SENS condition as defined in Chapter 7 implies that there are G-programs of the n-input AND of polynomial length. However, it is not known whether the converse implication holds. In particular, no super-polynomial lower bound for the size of G-programs for the AND-function in finite solvable groups is known.

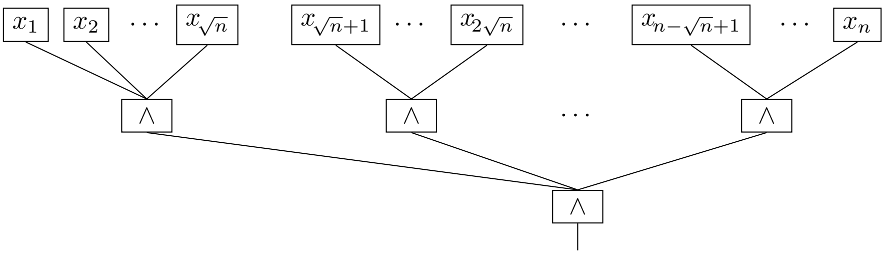

Figure 8.1: Building an n-input AND out of $\sqrt{n}$ many $\sqrt{n}$-input ANDs.

8.3 Proof outline

The proofs of Theorem 8.1 and Corollary 8.3 are based on the following observation: Assume there is a circuit of depth two and size 2^n for the n-input AND. Since the n-input AND can be decomposed as $\sqrt{n}$-input AND of $\sqrt{n}$ many $\sqrt{n}$-input ANDs (see Fig. 8.1), we obtain a CC^0 circuit[33] of depth 4 and size roughly $2^{\sqrt{n}}$. For proving Theorem 8.1, we use this to encode the C-Coloring-problem into the group:

Reducing C-Coloring. A C-coloring for $C \in \mathbb{N}$ of a graph $\Gamma = (V, E)$ is a map $\chi : V \to [1 \mathinner{..} C]$. A coloring χ is called *valid* if $\chi(u) \neq \chi(v)$ whenever $\{u, v\} \in E$. The C-Coloring problem is as follows: given an undirected graph $\Gamma = (V, E)$, the question is whether there is a valid C-coloring of Γ. By [CFK+15, Thm. 14.6], C-Coloring cannot be solved in time $2^{o(|V|+|E|)}$ unless ETH fails. The main technical statement for the reduction of C-Coloring is:

Theorem 8.6. *Let G be a finite solvable group of Fitting length three and assume there are normal subgroups $K \trianglelefteq H \trianglelefteq G$ such that* $\mathrm{FitLen}(K) = 2$*, $\mathcal{U}_2 G \leq H$, $|G/H| \geq 3$, and*

(I) for all $g \in G \setminus H$ we have $\eta_g(K) = K$,

(II) for all $h \in H$ we have $\mathrm{FitLen}(\eta_h(K)) \leq 1$.

Then EQN-SAT(G) and EQN-ID(G) cannot be decided in deterministic time $2^{o(\log^2 N)}$ under ETH where N is the length of the input expression.

The reduction. Given a graph Γ with n vertices and m edges, we construct an expression δ and an element $\tilde{h} \in G$ such that

(A) the length of δ is in $2^{\mathcal{O}(\sqrt{m+n})}$,

(B) δ can be computed in time polynomial in its length,

(C) $\delta = \tilde{h}$ is satisfiable if and only if Γ has a valid C-coloring, and

(D) $\sigma(\delta) = 1$ holds for all assignments σ if and only if Γ does *not* have a valid C-coloring.

For the number of colors we use $C = |G/H|$. Let N denote the input length for EQN-SAT (resp. EQN-ID). A $2^{o(\log^2 N)}$-time algorithm for EQN-SAT (resp. EQN-ID) would imply a $2^{o(n+m)}$-time algorithm for C-Coloring – contradicting ETH. Hence, it suffices to show points (A)–(D).

[33] CC^0 is the class of constant-depth, polynomial-size circuits using only MOD_m gates for some fixed m.

In order to construct the expression δ, we assign a variable X_i to every vertex v_i of Γ. Every assignment σ to the variables X_i yields a coloring χ_σ of Γ. During the proof, we also introduce some auxiliary variables in such a way that an assignment σ to the variables X_i can be extended to a satisfying assignment for $\delta = \tilde{h}$ if and only if χ_σ is a valid coloring of Γ.

We start by grouping the edges into roughly $\sqrt{m}$ batches of $\sqrt{m}$ edges each. For each batch of edges r, we construct an expression γ_r such that for every assignment σ to the variables X_i we have

- if χ_σ assigns the same color to two endpoints of an edge in the batch r, then for every assignment to the auxiliary variables, γ_r evaluates to an element of $\mathcal{U}_1 K$,
- otherwise, for every element $h \in K$, there is an assignment to the auxiliary variables such that γ_r evaluates to h.

The expression δ combines all the γ_r as an iterated commutator such that if one of the γ_r evaluates to something in $\mathcal{U}_1 K$, then δ evaluates to 1, and, otherwise, there is some assignment to the auxiliary variables such that δ evaluates to the fixed element $\tilde{h}$.

List of publications summarized in this Habilitationsschrift

[1] Laurent Bartholdi, Michael Figelius, Markus Lohrey, and Armin Weiß. Groups with ALOGTIME-hard word problems and PSPACE-complete circuit value problems. In *35th Computational Complexity Conference, CCC 2020, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 169 of *LIPIcs*, pages 29:1–29:29. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. Technical report at `https://arxiv.org/abs/1909.13781`. `doi:10.4230/LIPIcs.CCC.2020.29`.

[2] Markus Lohrey and Armin Weiß. The power word problem. In *44th International Symposium on Mathematical Foundations of Computer Science, MFCS 2019, Proceedings*, volume 138 of *LIPIcs*, pages 43:1–43:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. Technical report at `https://arxiv.org/abs/1904.08343`. `doi:10.4230/LIPIcs.MFCS.2019.43`.

[3] Alexei Miasnikov, Svetla Vassileva, and Armin Weiß. The conjugacy problem in free solvable groups and wreath products of abelian groups is in TC^0. *Theory of Computing Systems*, 63(4):809–832, 2019. Conference version at *CSR 2017* (best paper award). `doi:10.1007/s00224-018-9849-2`.

[4] Alexei G. Myasnikov and Armin Weiß. TC^0 circuits for algorithmic problems in nilpotent groups. In *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS 2017, Proceedings*, volume 83 of *LIPIcs*, pages 23:1–23:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017. Technical report at `http://arxiv.org/abs/1702.06616`, Journal version submitted to *International Journal of Algebra and Computation.* `doi:10.4230/LIPIcs.MFCS.2017.23`.

[5] Armin Weiß. A logspace solution to the word and conjugacy problem of generalized Baumslag-Solitar groups. In *Algebra and Computer Science*, volume 677 of *Contemporary Mathematics*, pages 185–212. American Mathematical Society, 2016. Technical report at `https://arxiv.org/abs/1602.02445`. `doi:10.1090/conm/677`.

[6] Armin Weiß. Hardness of equations over finite solvable groups under the exponential time hypothesis. In *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 102:1–102:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. `doi:10.4230/LIPIcs.ICALP.2020.102`.

Bibliography

[Ans76a] Michael Anshel. The conjugacy problem for HNN groups and the word problem for commutative semigroups. *Proc. Amer. Math. Soc.*, 61(2):223–224, 1976.

[Ans76b] Michael Anshel. Conjugate powers in HNN groups. *Proc. Amer. Math. Soc.*, 54:19–23, 1976.

[Ans76c] Michael Anshel. Decision problems for HNN groups and vector addition systems. *Mathematics of Computation*, 30(133):154–156, 1976.

[AS74] Michael Anshel and Peter Stebe. The solvability of the conjugacy problem for certain HNN groups. *Bull. Amer. Math. Soc.*, 80:266–270, 1974.

[Bar89] David A. Mix Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. *Journal of Computer and System Sciences*, 38(1):150–164, 1989. A preliminary version is in STOC 1986.

[BBR94] David A. Mix Barrington, Richard Beigel, and Steven Rudich. Representing Boolean functions as polynomials modulo composite numbers. *Computational Complexity*, 4:367–382, 1994.

[BCS92] Daniel P. Bovet, Pierluigi Crescenzi, and Riccardo Silvestri. A uniform approach to define complexity classes. *Theoretical Computer Science*, 104(2):263–283, 1992.

[Bee11] Benjamin Beeker. *Problèmes géométriques et algorithmiques dans des graphes de groupes.* PhD thesis, Université de Caen Basse-Normandie, 2011.

[BFLW19] Laurent Bartholdi, Michael Figelius, Markus Lohrey, and Armin Weiß. Groups with ALOGTIME-hard word problems and PSPACE-complete compressed word problems. *arXiv eprints*, abs/1909.13781, 2019.

[BG92] Richard Beigel and John Gill. Counting classes: Thresholds, parity, mods, and fewness. *Theoretical Computer Science*, 103(1):3–23, 1992.

[BGv03] Laurent Bartholdi, Rostislav I. Grigorchuk, and Zoran Šunik̀. Branch groups. In *Handbook of algebra, Vol. 3*, volume 3 of *Handb. Algebr.*, pages 989–1112. Elsevier/North-Holland, Amsterdam, 2003.

[BL04] Stanley Burris and J. Lawrence. Results on the equivalence problem for finite groups. *Algebra Universalis*, 52(4):495–500 (2005), 2004.

[Bla65] Norman Blackburn. Conjugacy in nilpotent groups. *Proceedings of the American Mathematical Society*, 16(1):143–148, 1965.

[BMM$^+$00] David A. Mix Barrington, Pierre McKenzie, Cristopher Moore, Pascal Tesson, and Denis Thérien. Equation satisfiability and program satisfiability for finite monoids. In *Mathematical Foundations of Computer Science 2000, 25th International Symposium,* *MFCS 2000, Proceedings*, volume 1893 of *Lecture Notes in Computer Science*, pages 172–181. Springer, 2000.

[BMPT97] Martin Beaudry, Pierre McKenzie, Pierre Péladeau, and Denis Thérien. Finite monoids: From word to circuit evaluation. *SIAM Journal on Computing*, 26(1):138–152, 1997.

[BN08] Laurent Bartholdi and Volodymyr V. Nekrashevych. Iterated monodromy groups of quadratic polynomials. I. *Groups, Geometry, and Dynamics*, 2(3):309–336, 2008.

[Boo59] W. W. Boone. The Word Problem. *Ann. of Math.*, 70(2):207–265, 1959.

[BS62] Gilbert Baumslag and Donald Solitar. Some two-generator one-relator non-Hopfian groups. *Bull. Amer. Math. Soc.*, 68:199–201, 1962.

[BST90] David A. Mix Barrington, Howard Straubing, and Denis Thérien. Non-uniform automata over groups. *Inf. Comput.*, 89(2):109–132, 1990.

[BT88] David A. Mix Barrington and Denis Thérien. Finite monoids and the fine structure of NC^1. *Journal of the ACM*, 35:941–952, 1988.

[CFK$^+$15] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms.* Springer, 2015.

[CFP96] John W. Cannon, William J. Floyd, and Walter R. Parry. Introductory notes on Richard Thompson's groups. *L'Enseignement Mathématique*, 42(3):215–256, 1996.

[CJ12] Matthew J. Craven and Henri C. Jimbo. Evolutionary algorithm solution of the multiple conjugacy search problem in groups, and its applications to cryptography. *Groups Complexity Cryptology*, 4:135–165, 2012.

[CLM76] E. Cardoza, R. Lipton, and A. R. Meyer. Exponential space complete problems for Petri nets and commutative semigroups: preliminary report. In *Eighth Annual ACM Symposium on Theory of Computing (Hershey, Pa., 1976)*, pages 50–54. Assoc. Comput. Mach., New York, 1976.

[CM87] Stephen A Cook and Pierre McKenzie. Problems complete for deterministic logarithmic space. *Journal of Algorithms*, 8(3):385–394, 1987.

[DE17] Volker Diekert and Murray Elder. Solutions of twisted word equations, EDT0L languages, and context-free groups. In *ICALP 2017, Proceedings*, volume 80 of *LIPIcs*, pages 96:1–96:14, Dagstuhl, Germany, 2017.

[Deh11] Max Dehn. Ueber unendliche diskontinuierliche Gruppen. *Math. Ann.*, 71:116–144, 1911.

[DMW16] Volker Diekert, Alexei G. Myasnikov, and Armin Weiß. Conjugacy in Baumslag's group, generic case complexity, and division in power circuits. *Algorithmica*, 74:961–988, 2016.

[DW17] Volker Diekert and Armin Weiß. Context-Free Groups and Bass-Serre Theory. In Juan González-Meneses, Martin Lustig, and Enric Ventura, editors, *Algorithmic and Geometric Topics Around Free Groups and Automorphisms*, Advanced Courses in Mathematics - CRM Barcelona. Birkhäuser, Basel, Switzerland, 2017.

[EK04] B. Eick and D. Kahrobaei. Polycyclic groups: A new platform for cryptology? *ArXiv Mathematics e-prints*, 2004.

[FGLZ20] Michael Figelius, Moses Ganardi, Markus Lohrey, and Georg Zetzsche. The complexity of knapsack problems in wreath products. In *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, pages 126:1–126:18, 2020.

[FH20] Attila Földvári and Gábor Horváth. The complexity of the equation solvability and equivalence problems over finite groups. *International Journal of Algebra and Computation*, 30(03):607–623, 2020.

[Föl17] Attila Földvári. The complexity of the equation solvability problem over semipattern groups. *IJAC*, 27(2):259, 2017.

[For03] Max Forester. On uniqueness of JSJ decompositions of finitely generated groups. *Comment. Math. Helv.*, 78(4):740–751, 2003.

[Ge93] Guoqiang Ge. Testing equalities of multiplicative representations in polynomial time (extended abstract). In *Proceedings of the 34th Annual Symposium on Foundations of Computer Science, FOCS 1993*, pages 422–426, 1993.

[GKLZ18] Moses Ganardi, Daniel König, Markus Lohrey, and Georg Zetzsche. Knapsack problems for wreath products. In *Proceedings of the 35th Symposium on Theoretical Aspects of Computer Science, STACS 2018*, volume 96 of *LIPIcs*, pages 32:1–32:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

[GMO20] Albert Garreta, Alexei Miasnikov, and Denis Ovchinnikov. Diophantine problems in solvable groups. *Bull. Math. Sci.*, 10(1):2050005, 27, 2020.

[GR02] Mikael Goldmann and Alexander Russell. The complexity of solving equations over finite groups. *Inf. Comput.*, 178(1):253–262, 2002.

[Gri80] Rostislav I. Grigorchuk. Burnside's problem on periodic groups. *Functional Analysis and Its Applications*, 14:41–43, 1980.

[GS83] Narain Gupta and Saïd Sidki. On the Burnside problem for periodic groups. *Mathematische Zeitschrift*, 182(3):385–388, 1983.

[GS99] Victor S. Guba and Mark V. Sapir. On subgroups of the R. Thompson group F and other diagram groups. *Matematicheskii Sbornik*, 190(8):3–60, 1999.

[GS07] Yuri Gurevich and Paul Schupp. Membership problem for the modular group. *SIAM J. Comput.*, 37:425–459, 2007.

[GS09] Dima Grigoriev and Vladimir Shpilrain. Authentication from matrix conjugation. *Groups Complexity Cryptology*, 1:199–205, 2009.

[Gv06] Rostislav I. Grigorchuk and Zoran Šunić. Asymptotic aspects of Schreier graphs and Hanoi Towers groups. *C. R. Math. Acad. Sci. Paris*, 342(8):545–550, 2006.

[GZ91] Max Garzon and Yechezkel Zalcstein. The complexity of Grigorchuk groups with application to cryptography. *Theoretical Computer Science*, 88(1):83–98, 1991.

[HAB02] William Hesse, Eric Allender, and David A. Mix Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *Journal of Computer and System Sciences*, 65:695–716, 2002.

[Hal69] Philip Hall. *The Edmonton notes on nilpotent groups.* Queen Mary College Mathematics Notes. Mathematics Department, Queen Mary College, London, 1969.

[Her94] Ulrich Hertrampf. *Über Komplexitätsklassen, die mit Hilfe von k-wertigen Funktionen definiert werden.* Habilitationsschrift, Universität Würzburg, 1994.

[Her97] Ulrich Hertrampf. The shapes of trees. In *Proceedings of COCOON 1997*, volume 1276 of *Lecture Notes in Computer Science*, pages 412–421. Springer, 1997.

[Her00] Ulrich Hertrampf. Algebraic acceptance mechanisms for polynomial time machines. *SIGACT News*, 31(2):22–33, 2000.

[Hes01] William Hesse. Division is in uniform TC^0. In Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen, editors, *ICALP*, volume 2076 of *Lecture Notes in Computer Science*, pages 104–114. Springer, 2001.

[HF94] K. J. Horadam and G. E. Farr. The conjugacy problem for HNN extensions with infinite cyclic associated groups. *Proc. Amer. Math. Soc.*, 120(4):1009–1015, 1994.

[HLMS07] Gábor Horváth, John Lawrence, László Mérai, and Csaba Szabó. The complexity of the equivalence problem for nonsolvable groups. *Bull. Lond. Math. Soc.*, 39(3):433–438, 2007.

[HLS$^+$93] Ulrich Hertrampf, Clemens Lautemann, Thomas Schwentick, Heribert Vollmer, and Klaus W. Wagner. On the power of polynomial time bit-reductions. In *Proceedings of the Eighth Annual Structure in Complexity Theory Conference*, pages 200–207. IEEE Computer Society Press, 1993.

[HLS19] Derek F. Holt, Markus Lohrey, and Saul Schleimer. Compressed decision problems in hyperbolic groups. In *Proceedings of STACS 2019*, volume 126 of *LIPIcs*, pages 37:1–37:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[Hol00] D. Holt. Word-hyperbolic groups have real-time word problem. *Int. J. Algebr. Comput.*, 10:221–227, 2000.

[Hor81] K. J. Horadam. The word problem and related results for graph product groups. *Proc. American Mathematical Society*, 82:407–408, 1981.

[Hor84] K. J. Horadam. The conjugacy problem for graph products with central cyclic edge groups. *Proc. Amer. Math. Soc.*, 91(3):345–350, 1984.

[Hor11] Gábor Horváth. The complexity of the equivalence and equation solvability problems over nilpotent rings and groups. *Algebra Universalis*, 66(4):391–403, 2011.

[Hor15] Gábor Horváth. The complexity of the equivalence and equation solvability problems over meta-Abelian groups. *J. Algebra*, 433:208–230, 2015.

[HS06] Gábor Horváth and Csaba A. Szabó. The complexity of checking identities over finite groups. *IJAC*, 16(5):931–940, 2006.

[HS11] Gábor Horváth and Csaba Szabó. The extended equivalence and equation solvability problems for groups. *Discrete Math. Theor. Comput. Sci.*, 13(4):23–32, 2011.

[HS12] Gábor Horváth and Csaba Szabó. Equivalence and equation solvability problems for the alternating group $\mathbf{A}_4$. *J. Pure Appl. Algebra*, 216(10):2170–2176, 2012.

[Hup67] B. Huppert. *Endliche Gruppen. I.* Die Grundlehren der Mathematischen Wissenschaften, Band 134. Springer-Verlag, Berlin-New York, 1967.

[HVW96] Ulrich Hertrampf, Heribert Vollmer, and Klaus Wagner. On balanced versus unbalanced computation trees. *Mathematical Systems Theory*, 29(4):411–421, 1996.

[IJCR91] Oscar H. Ibarra, Tao Jiang, Jik H. Chang, and Bala Ravikumar. Some classes of languages in NC^1. *Inform. and Comput.*, 90(1):86–106, 1991.

[IKK20] Pawel M. Idziak, Piotr Kawalek, and Jacek Krzaczkowski. Intermediate problems in modular circuits satisfiability. In *LICS 2020, Proceedings*. ACM, 2020. Preprint at `https://arxiv.org/abs/2002.08626`.

[IKKW20] Paweł Idziak, Piotr Kawałek, Jacek Krzaczkowski, and Armin Weiß. Equation satisfiability in solvable groups. *arXiv eprints*, abs/2010.11788, 2020. Accepted at TOCS with minor revision.

[IPZ01] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001.

[JLM] Birgit Jenner, Klaus-Jörn Lange, and Pierre McKenzie. Tree isomorphism and some other complete problems for deterministic logspace. publication #1059, DIRO, Université de Montréal, 1997.

[JMT96] Birgit Jenner, Pierre McKenzie, and Denis Thérien. Logspace and logtime leaf languages. *Information and Computation*, 129(1):21–33, 1996.

[Kau17] Jonathan Kausch. *The parallel complexity of certain algorithmic problems in group theory*. Dissertation, Institut für Formale Methoden der Informatik, Universität Stuttgart, 2017.

[Kha81] O. G. Kharlampovich. A finitely presented solvable group with unsolvable word problem. *Izv. Akad. Nauk SSSR Ser. Mat.*, 45(4):852–873, 928, 1981.

[KL18a] Daniel König and Markus Lohrey. Evaluation of circuits over nilpotent and polycyclic groups. *Algorithmica*, 80(5):1459–1492, 2018.

[KL18b] Daniel König and Markus Lohrey. Parallel identity testing for skew circuits with big powers and applications. *IJAC*, 28(6):979–1004, 2018.

[KLC$^+$00] Ki Hyoung Ko, Sang Jin Lee, Jung Hee Cheon, Jae Woo Han, Ju-sung Kang, and Choonsik Park. New public-key cryptosystem using braid groups. In *Advances in cryptology—CRYPTO 2000 (Santa Barbara, CA)*, volume 1880 of *Lecture Notes in Comput. Sci.*, pages 166–183. Springer, Berlin, 2000.

[KLR07] Andreas Krebs, Klaus-Jörn Lange, and Stephanie Reifferscheid. Characterizing TC^0 in terms of infinite groups. *Theory Comput. Syst.*, 40(4):303–325, 2007.

[KM79] M. I. Kargapolov and Ju. I. Merzljakov. *Fundamentals of the theory of groups*, volume 62 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1979. Translated from the second Russian edition by Robert G. Burns.

[Kom19] Michael Kompatscher. Notes on extended equation solvability and identity checking for groups. *Acta Math. Hungar.*, 159(1):246–256, 2019.

[KR66] M. I. Kargapolov and V. N. Remeslennikov. The conjugacy problem for free solvable groups. *Algebra i Logika Sem.*, 5(6):15–25, 1966.

[Kro90] Peter H. Kropholler. Baumslag-Solitar groups and some other groups of cohomological dimension two. *Comment. Math. Helv.*, 65(4):547–558, 1990.

[KRR$^+$69] M. I. Kargapolov, V. N. Remeslennikov, N. S. Romanovskii, V. A. Roman'kov, and V. A. Čurkin. Algorithmic questions for σ-powered groups. *Algebra i Logika*, 8:643–659, 1969.

[KTT07] Ondrej Klíma, Pascal Tesson, and Denis Thérien. Dichotomies in the complexity of solving systems of equations over finite semigroups. *Theory Comput. Syst.*, 40(3):263–297, 2007.

[Lau12] Jürn Laun. *Solving algorithmic problems in Baumslag-Solitar groups and their extensions using data compression.* Dissertation, Institut für Formale Methoden der Informatik, Universität Stuttgart, 2012.

[Loc92] Jody Meyer Lockhart. The conjugacy problem for graph products with infinite cyclic edge groups. *Proc. Amer. Math. Soc.*, 114(3):603–606, 1992.

[Loh05] Markus Lohrey. Decidability and complexity in automatic monoids. *International Journal of Foundations of Computer Science*, 16(4):707–722, 2005.

[Loh06] Markus Lohrey. Word problems and membership problems on compressed words. *SIAM J. Comput.*, 35(5):1210–1240, 2006.

[Loh14] Markus Lohrey. *The Compressed Word Problem for Groups.* Springer Briefs in Mathematics. Springer, 2014.

[Lot83] M. Lothaire. *Combinatorics on Words*, volume 17 of *Encyclopedia of Mathematics and Its Applications.* Addison-Wesley, 1983. Reprinted by *Cambridge University Press*, 1997.

[LS06] Markus Lohrey and Géraud Sénizergues. Theories of HNN-extensions and amalgamated products. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *ICALP*, volume 4052 of *Lecture Notes in Computer Science*, pages 504–515. Springer, 2006.

[LS07] Jörg Lehnert and Pascal Schweitzer. The co-word problem for the Higman-Thompson group is context-free. *Bulletin of the London Mathematical Society*, 39(2):235–241, 02 2007.

[LZ77] Richard J. Lipton and Yechezkel Zalcstein. Word problems solvable in logspace. *Journal of the ACM*, 24:522–526, 1977.

[LZ18] Markus Lohrey and Georg Zetzsche. Knapsack in graph groups. *Theory of Computing Systems*, 62(1):192–246, 2018.

[Mag39] Wilhelm Magnus. On a theorem of Marshall Hall. *Ann. of Math. (2)*, 40:764–768, 1939.

[Mak77] Gennadií Semyonovich Makanin. The problem of solvability of equations in a free semigroup. *Math. Sbornik*, 103:147–236, 1977. English transl. in Math. USSR Sbornik 32 (1977).

[Mak84] Gennadií Semyonovich Makanin. Decidability of the universal and positive theories of a free group. *Izv. Akad. Nauk SSSR*, Ser. Mat. 48:735–749, 1984. In Russian; English translation in: *Math. USSR Izvestija, 25*, 75–88, 1985.

[Mal58] Anatolij I. Mal'cev. On homomorphisms of finite groups. *Ivano Gosudarstvennyi Pedagogicheskii Institut Uchenye Zapiski*, 18:49–60, 1958.

[Mat66] J. Matthews. The conjugacy problem in wreath products and free metabelian groups. *Trans. Amer. Math. Soc.*, 121:329–339, 1966.

[Mil71] C. F. Miller III. *On group-theoretic decision problems and their classification*, volume 68 of *Annals of Mathematics Studies*. Princeton University Press, 1971.

[MM82] Ernst W. Mayr and Albert R. Meyer. The complexity of the word problems for commutative semigroups and polynomial ideals. *Advances in Math.*, 46:305–329, 1982.

[MMNV15] Jeremy Macdonald, Alexei G. Myasnikov, Andrey Nikolaev, and Svetla Vassileva. Logspace and compressed-word computations in nilpotent groups. *arXiv eprints*, abs/1503.03888, 2015.

[MNU15] Alexei Myasnikov, Andrey Nikolaev, and Alexander Ushakov. Knapsack problems in groups. *Mathematics of Computation*, 84:987–1016, 2015.

[Mos66] A. Mostowski. Computational algorithms for deciding some problems for nilpotent groups. *Fundamenta Mathematicae*, 59(2):137–152, 1966.

[MRUV10] A. Myasnikov, V. Roman'kov, A. Ushakov, and A. Vershik. The word and geodesic problems in free solvable groups. *Trans. Amer. Math. Soc.*, 362:4655–4682, 2010.

[MT98] Alexis Maciel and Denis Thérien. Threshold circuits of small majority-depth. *Inf. Comput.*, 146(1):55–83, 1998.

[MVW17] Alexei Miasnikov, Svetla Vassileva, and Armin Weiß. The conjugacy problem in free solvable groups and wreath product of abelian groups is in TC^0. In Pascal Weil, editor, *Computer Science - Theory and Applications - 12th International*

Computer Science Symposium in Russia, CSR 2017, Kazan, Russia, June 8-12, 2017, Proceedings, volume 10304 of *Lecture Notes in Computer Science*, pages 217–231. Springer, 2017.

[MVW18] Alexei Myasnikov, Svetla Vassileva, and Armin Weiß. Log-space complexity of the conjugacy problem in wreath products. In *Infinite Group Theory*, chapter 12, pages 215–236. World Scientific, 2018.

[Nek05] Volodymyr Nekrashevych. *Self-similar groups*, volume 117 of *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI, 2005.

[Nov55] P. S. Novikov. On the algorithmic unsolvability of the word problem in group theory. *Trudy Mat. Inst. Steklov*, pages 1–143, 1955. In Russian.

[Pap94] Christos H. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994.

[Rei08] Omer Reingold. Undirected connectivity in log-space. *J. ACM*, 55(4):17:1–17:24, September 2008.

[Rem73] V. N. Remeslennikov. An example of a group, finitely presented in the variety $\mathfrak{A}^5$, with an undecidable word problem. *Algebra i Logika*, 12:577–602, 618, 1973.

[Rob93] David Robinson. *Parallel Algorithms for Group Word Problems.* PhD thesis, University of California, San Diego, 1993.

[Rom79] Vitaly Roman'kov. Equations in free metabelian groups. *Siberian Mathematical Journal*, 20, 05 1979.

[RS70] V.N. Remeslennikov and V. G. Sokolov. Certain properties of the Magnus embedding. *Algebra i logika*, 9(5):566–578, 1970.

[Sed17] Sidi Mohamed Sedjelmaci. Two fast parallel GCD algorithms of many integers. In Michael A. Burr, Chee K. Yap, and Mohab Safey El Din, editors, *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation, ISSAC 2017, Kaiserslautern, Germany, July 25-28, 2017*, pages 397–404. ACM, 2017.

[Ser80] Jean-Pierre Serre. *Trees.* Springer, 1980. French original 1977.

[Sim79] Hans-Ulrich Simon. Word problems for groups and contextfree recognition. In *Proceedings of Fundamentals of Computation Theory (FCT'79), Berlin/Wendisch-Rietz (GDR)*, pages 417–422. Akademie-Verlag, 1979.

[Smo87] Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, pages 77–82, 1987.

[SS06] Steve Seif and Csaba Szabó. Computational complexity of checking identities in 0-simple semigroups and matrix semigroups over finite fields. *Semigroup Forum*, 72(2):207–222, 2006.

[SY10] Amir Shpilka and Amir Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science*, 5(3-4):207–388, 2010.

[SZ06] V. Shpilrain and G. Zapata. Combinatorial group theory and public key cryptography. *Applicable Algebra in Engineering, Communication and Computing*, 17:291–302, 2006.

[Tit72] Jacques Tits. Free subgroups in linear groups. *Journal of Algebra*, 20(2):250–270, 1972.

[Vas11] Svetla Vassileva. Polynomial time conjugacy in wreath products and free solvable groups. *Groups Complexity Cryptology*, 3(1):105–120, 2011.

[Vol99] Heribert Vollmer. *Introduction to Circuit Complexity.* Springer, Berlin, 1999.

[Waa81] Stephan Waack. Tape complexity of word problems. In *FCT 1981, Proceedings*, volume 117 of *Lecture Notes in Computer Science*, pages 467–471. Springer, 1981.

[Waa90] Stephan Waack. The parallel complexity of some constructions in combinatorial group theory. *Journal of Information Processing and Cybernetics*, 26(5-6):265–281, 1990.

[Wei15] Armin Weiß. *On the Complexity of Conjugacy in Amalgamated Products and HNN Extensions.* Dissertation, Institut für Formale Methoden der Informatik, Universität Stuttgart, 2015.

[WW20] Jan Philipp Wächter and Armin Weiß. An automaton group with PSPACE-complete word problem. In *37th International Symposium on Theoretical Aspects of Computer Science, STACS 2020, March 10-13, 2020, Montpellier, France*, pages 6:1–6:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020.

[WWC+11] Lihua Wang, Licheng Wang, Zhenfu Cao, Eiji Okamoto, and Jun Shao. New constructions of public-key encryption schemes from conjugacy search problems. In *Information security and cryptology*, volume 6584 of *Lecture Notes in Comput. Sci.*, pages 1–17. Springer, Heidelberg, 2011.