Seminar: Decision problems in groups

C. Löh/M. Uschold/F. Hofmann July 2023

Many group-theoretic problems look deceivingly simple, but turn out to be algorithmically unsolvable:

Given a presentation of a group, can we decide whether the resulting group is trivial, finite, cyclic? Which order does a given element have? Are two given elements conjugate?

The algorithmic unsolvability of these problems has far-reaching consequences for other fields; e.g., many decision problems in geometry and topology can be reduced to such undecidable problems in group theory and thus are also algorithmically undecidable.

In this seminar, we will introduce the language of Turing machines, (un)decidability, and classical undecidable problems. Moreover, we will learn the basics on presentations of groups and group-theoretic constructions. We will then combine both aspects and study the (un)decidability of various problems in group theory. In the last part of the seminar, we will see some consequences of this in topology. Conversely, we will also study certain special classes of groups in which several of the problems do have algorithmic solutions.

Several talks will also be suitable for students in the Lehramt Gymnasium track.

References. Large parts of the seminar will follow the book by Rotman [Rot95]. However, you may and should also consult other sources.

Prerequisites. Basic knowledge about groups will be assumed. The talks on applications in topology require some basic (algebraic) topology background.

Admin and preparation. Please take the general advice on seminars into account: https://loeh.app.uni-regensburg.de/teaching/seminar_preparation.pdf

Organisers.
Prof. Dr. Clara Löh (clara.loeh@ur.de),
Matthias Uschold (matthias.uschold@ur.de),
Franziska Hofmann (franziska2.hofmann@ur.de; starting from 10/2023)

Basics of Computability Theory

Talk 1 (Turing machines).
Main reference: [Rot95, p. 420–425] [Cut80, Section 4.1] [DW83, Chapter 6] [BJ74, Chapter 3]
We will capture computability via the notion of *Turing machines*.

- Define Turing machines and give easy examples.
- Define recursively enumerable and recursive sets and give examples.
- Prove (by a cardinality argument) that there must be subsets of $\mathbb{N}$ that are not recursive or recursively enumerable.
- Define Gödel numberings.
- There exist many minor modifications of Turing machines (e.g., with 42 bands instead of just one). Present some of them and sketch a proof that a modification of your choice leads to the same notion of recursive sets [DW83, Chapter 6.6].

Talk 2 (Church's Thesis and Turing completeness in computer games).
Main reference: [Cut80, Chapter 4] [DW83, Chapter 3] [Rot95, pp. 420–424] [BJ74, Chapters 6–8]
Turing machines are by far not the only approach to defining computability. In fact, there exist a bunch of modifications as well as entirely different models for defining computability. It turns out that many of these notions define the same notion of computability. In this case, we call such a system *Turing complete.*

Turing machines and other notions intend to be Turing- complete. However, there are also (computer) games that turn out to be Turing-complete by accident.

- Recall the definition of Turing machines.
- Present a different approach from theoretical computer science, (e.g., λ-calculus, recursion theory).
- Present an approach to computation or recognition of abstract languages that is *not* Turing complete (e.g., regular expressions).
- State that most programming languages are Turing complete (e.g., Python, C, Haskell, but also TEX).
- Pick your favourite Turing complete computer game or card game, describe the available building blocks and sketch a method for proving that the game is Turing complete [CBH19, Ric].

Talk 3 (the Halting Problem and Rice's Theorem).
Main reference: [DW83, Chapter 4.2, 4.7] [Rot95, pp. 420–424] [BJ74, Chapter 5]
We can define a subset of the set of Turing machines and inputs that stops on the given input after a finite number of steps, the *Halting Problem.* Using a Gödel numbering, this is a subset of $\mathbb{N}$ that is not recursive.

- Motivate why the Halting Problem and the problem posed in Rice's Theorem might be interesting from a programmer's point of view. You may use here that many programming languages are Turing complete (see Talk 2).
- Formulate a precise statement for the unsolvability of the Halting Problem.
- Prove that the Halting Problem set in $\mathbb{N}$ is not recursive.
- Prove Rice's Theorem: all non-trivial properties involving semantics of Turing machines are undecidable.

Basics of Group Theory

Talk 4 (presentations of groups and Cayley graphs).
Main reference: [Löh17, Chapter 2.2, 2.3, 3.2]

- Define free groups.
- Introduce groups given by generators and relations, finitely generated, and finitely presented groups.

- Define the following constructions: free products, free amalgamated products.
- State concrete constructions as well as universal properties of these constructions.
- Illustrate these constructions with examples
- Define Cayley graphs.

Talk 5 (HNN extensions).
Main reference: [Löh17, Section 2.3.2] [Rot95, p. 401–417 (without the geometric parts)]

- Define HNN extensions and give examples.
- Prove that every countable group can be embedded into a group generated by two generators.
- Prove that there are uncountably many isomorphism types of finitely generated groups.

The word problem

Talk 6 (the word problem).
Main reference: [Mei08, Section 5.2]
The word problem asks if a given word in the generators of a presentation of a group represents the trivial element. In this talk, we establish solvability of the word problem for some classes of groups.

- Define the word problem for finitely generated groups with a set of generators.
- Show that the solvability of the word problem does not depend on the choice of a finite generating set [Mil92, Lemma 2.2]. We can thus talk about the solvability of the word problem of a finitely generated group without specifying the generating set.
- Show that finite groups, free abelian groups $\mathbb{Z}^n$, and free groups F_n have solvable word problem (by describing an algorithm solving the word problem; note that you may fix a finite generating set of your choice).
- Define residual finiteness and show that finitely presented, residually finite groups have solvable word problem [Mil92, Theorem 5.3]. Give examples of such groups (e.g., matrix groups).

Talk 7 (Dehn presentations and hyperbolic groups).
Main reference: [Löh17, Chapter 7.4] [BH99, Chapter III.2]
Some groups admit 'nice' presentations, implying the solvability of the word problem. This class of groups contains all hyperbolic groups.

- Define Dehn presentations.

- Show that groups given by Dehn presentations have solvable word problem.
- Give a short overview of important properties and examplex of hyperbolic groups [Löh17, Chapters 7.3, 7.5].
- Sketch a proof that hyperbolic groups admit Dehn presentations and thus have solvable word problem.

Talk 8 (One-relator groups).
Main reference: [MS73] [Put] [Mag32]
One-relator groups are groups admitting a presentation with a single relation, i.e., of the form $\langle s_1, \ldots, s_n \mid r_1 \rangle$. Surprisingly, many problems in this class of groups are solvable.

- Define one-relator groups and give examples, in particular, surface groups and Baumslag–Solitar groups $\mathrm{BS}(m, n)$.
- Give an explicit solution of the word problem for $\mathrm{BS}(1, 2)$ [Mei08, Proposition 5.4].
- Show that one-relator groups have solvable word problem. You may blackbox/sketch some intermediate results (e.g., the Freiheitssatz).
- Time permitting: Mention more results known for the class of one-relator groups (e.g., examples of hyperbolic one-relator groups).

Unsolvability of the word problem

Talk 9 (groups with unsolvable word problem).
Main reference: [Rot95, p. 425–433]

- Recall from Talk 3 that the Halting Problem is unsolvable.
- Recall the definition of the word problem of groups and adapt it to semigroups.
- Prove the Markov–Post theorem: There exists a finitely presented semigroup with unsolvable word problem.
- Prove the Novikov–Boone–Britton theorem: There exists a finitely presented *group* with unsolvable word problem. For this step, you may assume Boone's Lemma (which will be proved in Talk 10).

Talk 10 (Boone's lemma).
Main reference: [Rot95, p. 425–433]
In Talk 9, we proved the Novikov–Boone–Britton theorem, relying on Boone's lemma, which will be proved in this talk.

- Recall the statement of the Novikov–Boone–Britton theorem.
- Sketch the proof of Boone's lemma.
- Provide a geometric motivation for the algebraic proof.

Talk 11 (Higman's embedding theorem).
Main reference: [Rot95, p. 450–464]
Higman's embedding theorem states that every recursively presented group can be embedded into a finitely presented group.

- Define recursively presented groups.
- Give examples (e.g., finitely presented groups, Lamplighter groups).
- Sketch a proof of Higman's embedding theorem.

Talk 12 (Markov properties of groups).
Main reference: [Rot95, p. 464–470]
A large class of undecidable properties about groups are the *Markov* properties.

- Prove the existence of universal finitely presented groups.
- Define Markov properties.
- Give examples (e.g., being trivial, finite, abelian, ...)
- Prove the Adian–Rabin theorem: Markov properties cannot be decided by algorithms.

Undecidable problems in topology

We will apply the results from the last section to conclude that some problems in topology are not algorithmically decidable.

Talk 13 (homeomorphism problem for 4-manifolds).
Main reference: [MG21], and the references therein

- Recall the definition of the fundamental group.
- State homotopy invariance and the theorem of Seifert and van Kampen.
- Explain how to obtain a presentation of the fundamental group of a simplicial complex [Sti93, Chapter 4].
- Show that the following problem is *not* decidable: Given two 4-manifolds M and N, are M and N homeomorphic?

Talk 14 (second homology, knot groups).
Main reference: [Gor95]
State that the following problems are *not* decidable: For a finitely presented group G,

- decide if $H_2(G;\mathbb{Z}) \cong 0$;
- compute the deficiency of G;
- decide if G is a higher-dimensional knot group.

Prove at least one of these theorems (and introduce the necessary notions).

References

[BH99] Martin R. Bridson and André Haefliger. *Metric spaces of non-positive curvature*, volume 319 of *Grundlehren Math. Wiss.* Springer, 1999.

[BJ74] George Boolos and Richard Jeffrey. Computability and Logic. Cambridge University Press, 1974.

[CBH19] Alex Churchill, Stella Biderman, and Austin Herrick. Magic: The Gathering is Turing Complete, 2019. arXiv:1904.09828.

[Cut80] Nigel Cutland. Computability. An introduction to recursive function theory. Cambridge University Press, 1980.

[DW83] Martin D. Davis and Elaine J. Weyuker. Computability, complexity, and languages. Fundamentals of theoretical computer science. Computer Science and Applied Mathematics. Academic Press, 1983.

[Gor95] C. McA. Gordon. Some embedding theorems and undecidability questions for groups. In *Combinatorial and geometric group theory. Proceedings of a workshop held at Heriot-Watt University, Edinburgh, GB, spring of 1993*, pages 105–110. Cambridge University Press, 1995.

[Löh17] Clara Löh. *Geometric group theory. An introduction.* Universitext. Springer, 2017.

[Mag32] W. Magnus. Das Identitätsproblem für Gruppen mit einer definierenden Relation. *Math. Ann.*, 106:295–307, 1932.

[Mei08] John Meier. *Groups, graphs and trees. An introduction to the geometry of infinite groups.*, volume 73 of *Lond. Math. Soc. Stud. Texts.* Cambridge University Press, 2008.

[MG21] Cameron McA. Gordon. On the homeomorphism problem for 4-manifolds. *N. Z. J. Math.*, 52:821–826, 2021.

[Mil92] Charles F. III Miller. Decision problems for groups – survey and reflections. In *Algorithms and classification in combinatorial group theory. Lectures of a workshop on algorithms, word problems and classification in combinatorial group theory, held at MSRI, Berkeley, CA, USA , January 1989*, pages 1–59. Springer, 1992.

[MS73] James McCool and Paul E. Schupp. On one relator groups and HNN extensions. *J. Aust. Math. Soc.*, 16:249–256, 1973.

[Put] Andrew Putnam. One-relator groups. https://www3.nd.edu/~andyp/notes/OneRelator.pdf.

[Ric] Matt Rickard. Accidentally Turing Complete. https://matt-rickard.com/accidentally-turing-complete.

[Rot95] Joseph J. Rotman. *An introduction to the theory of groups.*, volume 148 of *Graduate Texts in Mathematics.* Springer, 4th ed. edition, 1995.

[Sti93] John Stillwell. *Classical topology and combinatorial group theory*, volume 72 of *Graduate Texts in Mathematics.* Springer, second edition, 1993.