

Seminar: Proof Lab: Simplicial Topology

C. Löh

July 2021

Many topological objects admit simple combinatorial descriptions through so-called simplicial complexes (a higher-dimensional version of graphs). This simplifies the computation of invariants from algebraic topology. Conversely, many situations arising in real-world applications can be modelled by simplicial complexes.

In this seminar, we will study combinatorial aspects of algebraic topology and their applications. Moreover, for additional excitement, we will implement fragments of the theory in a proof assistant. Proof assistants are not only used within mathematics, but also in industrial settings to verify properties of complex systems.

Talks

A short introduction to Lean

We give a very short introduction on how to formalise and prove mathematical statements in the proof assistant `Lean`. This is not a comprehensive course on the programming language `Lean`, but only an introduction for “the working mathematician”. The goal is to provide a setup for the implementation in the third part of the seminar. All participants should install `Lean 3` and a `git` client (no previous knowledge/experience concerning these tools is required).

Talk 1 (The `Lean` proof assistant). Representation of mathematical objects and proofs in `Lean`; a short explanation of the Curry-Howard correspondence; first, basic, examples.

Talk 2 (Further examples in `Lean`). Further modelling and proof techniques in `Lean`; how to use `mathlib`.

Simplicial complexes and their invariants

This part is the mathematical core of the seminar: We study simplicial complexes, examples of topological invariants of simplicial complexes, and example applications of simplicial topology.

Talk 3 (The category of simplicial complexes). Abstract simplicial complexes; simplicial maps; basic examples; link, star, wedge; joins; the face poset; barycentric and stellar subdivisions.

Literature: [6, Chapter 2.1] [11, Section 3]

Talk 4 (Geometric realisation). Geometric simplicial complexes; geometric realisation of (abstract) simplicial complexes; simplicial approximation theorem (including at least a sketch of proof).

Literature: [6, Chapter 2.2] [11, Sections 14–16 (without homology)]

Talk 5 (Applications: Mathematical modelling with simplicial complexes). Vietoris-Rips complexes; flag complexes; Čech complexes; nerves of covers; the nerve map (from a space to the nerve of an open cover); decision tasks and consensus.

Literature: [4, Chapter 2] [3, p. 355f]

Talk 6 (Simplicial homology). Construction of simplicial homology (with \mathbb{Z} -, \mathbb{R} -, \mathbb{F}_2 -coefficients) of simplicial complexes and simplicial maps; basic properties; very basic examples (further examples will be given in Talk 9).

Literature: [6, Chapter 3.2] [11, Sections 10, (11), 12]

Talk 7 (Topological invariance of simplicial homology). Recap of homeomorphisms, homotopy, and homotopy equivalences; invariance of simplicial homology under homeomorphisms and homotopy equivalences (including sketch proofs).

Literature: [11, Sections (13), 18, 19]

Talk 8 (Application: Sensor network coverage). Rips complexes (recap); sensor network coverage problems; homological criteria for sensor network coverage problems.

Literature: [4, Chapter 5.6] [5]

Talk 9 (Euler characteristic). Definition of the Euler characteristic of finite simplicial complexes; Euler-Poincaré formula; examples: spheres, tori, projective plane, Klein bottle, ... (both Euler characteristic and homology computations).

Literature: [6, Chapter 3.3.2] [11, Section 3]

Talk 10 (Application: Counting via topology). Configuration spaces of graphs; Euler calculus; target enumeration; enumerating vehicles; topological tomography.

Literature: [4, Chapter 3]

Proof lab: Formalisation of simplicial topology in Lean

In the final part of the seminar, we will implement fragments of the theory of simplicial complexes in `Lean`. We will do this together: it is expected that all participants contribute to the joint implementation.

Project 11 (Simplicial complexes in Lean). Formalisation of simplicial complexes in `Lean`; simple examples; simple properties;

Project 12 (Simplicial maps in Lean). Formalisation of simplicial maps in `Lean`; simple examples; simple properties;

Project 13 (Euler characteristic in Lean). Formalisation of the Euler characteristic in `Lean`; simple examples; simple properties;

Project 14 (Constructions and examples in Lean). More involved examples of simplicial complexes in `Lean`; further constructions of simplicial complexes, such as joins, subdivisions, etc., in `Lean`; example computations on simplicial complexes in `Lean`;

Ablauf des Seminars

Notwendig für den Scheinerwerb sind:

- Ein 80-minütiger Vortrag; die verbleibenden 10 Minuten der Sitzung werden wir für die Diskussion verwenden.
- Regelmäßige und aktive Teilnahme im Seminar (stellen Sie Fragen während der Vorträge, wenn Sie etwas nicht verstehen!).
- Ein Handout (Englisch) von ein bis zwei Seiten zu Ihrem Vortrag, das die wichtigsten Aspekte des Vortrags und ein paar kleine Übungsaufgaben für die anderen Teilnehmer enthält; diese Aufgaben sollen dazu anregen, sich nochmal mit den Inhalten des Vortrags zu beschäftigen.
- Prüfungsleistung: Eine schriftliche Ausarbeitung des Vortrags; diese sollte bis spätestens eine Woche vor dem Vortrag abgegeben werden. (Wird das Seminar als Proseminar oder als unbenotetes Seminar belegt, ist diese schriftliche Ausarbeitung nicht verpflichtend, aber dringend empfohlen.) Die Ausarbeitung kann auf Deutsch oder Englisch verfasst sein.
- Bitte kommen Sie spätestens zwei Wochen vor Ihrem Vortrag vorbei, um etwaige Fragen zu klären und den Vortrag durchzusprechen.
- Die Seminarleistungen werden wie in den entsprechenden Prüfungsordnungen/Modulkatalogen benotet und angerechnet.

Hinweise zur Vorbereitung

- Beginnen Sie frühzeitig mit der Vorbereitung (am besten vor Beginn des Semesters) und nutzen Sie Sprechstunden und sonstige Betreuungsangebote.
- Grundvoraussetzung für einen Seminarvortrag ist das mathematische Verständnis des Stoffes. Dabei sollten Sie mehr über das Thema wissen als Sie im Vortrag erwähnen werden.
- Geben Sie zu Beginn einen kurzen Überblick über Ihren Vortrag. Stellen Sie die Hauptaussagen Ihres Vortrags soweit wie möglich an den Anfang; damit vermeiden Sie es, diese am Ende des Vortrags unter Zeitdruck erläutern zu müssen.
- Unterscheiden Sie für das Publikum klar erkennbar zwischen Wichtigem und weniger Wichtigem. Überfordern Sie die Zuhörer nicht durch zuviele technische Details (Sie sollten diese aber selbstverständlich verstanden haben). Erklären Sie lieber die wesentlichen Ideen/Beweisschritte.
- Strukturieren Sie Ihren Vortrag; Überschriften für einzelne Abschnitte können dabei helfen. Je logischer und natürlicher Ihr Vortrag aufgebaut ist, desto leichter hält sich der Vortrag und desto verständlicher ist er.
- Machen Sie sich im Aufbau des Vortrags unabhängig von der Literatur. Ein Aufbau, der für einen Text sinnvoll ist, kann für einen Vortrag ungeeignet sein.

- Seien Sie der Literatur gegenüber kritisch. Sie sollten auch versuchen, selbst geeignete ergänzende Literatur zu finden. Geeignete Ausgangspunkte sind zum Beispiel:

<http://books.google.com>
<http://www.ams.org/mathscinet>
<http://www.springerlink.com>

- Planen Sie den zeitlichen Ablauf des Vortrags. Überlegen Sie sich schon vor dem Vortrag, welche Teile Sie bei Zeitnot kürzen können und welche Sie, wenn es die Zeit erlaubt, ausführlicher behandeln wollen. Ein Probevortrag kann helfen, den zeitlichen Ablauf des Vortrags abzuschätzen.
- Berücksichtigen Sie bei der Vorbereitung, was in den Vorträgen vor bzw. nach Ihrem eigenen Vortrag vorgesehen ist – im Zweifel sollten Sie sich mit den anderen Vortragenden absprechen, damit es nicht zu Lücken, Inkonsistenzen oder Überschneidungen kommt. Überlegen Sie, welche Begriffe/Aussagen aus den vorherigen Vorträgen Sie nochmal kurz wiederholen sollten.
- Sie können die Ausarbeitung und das Handout handschriftlich abgeben. Andererseits bieten die Ausarbeitung und das Handout aber auch eine gute Gelegenheit, das Textsatzsystem \LaTeX besser kennenzulernen [10]; dafür werden auch \LaTeX -Vorlagen zur Verfügung gestellt:
http://www.mathematik.uni-regensburg.de/loeh/teaching/cryptosem_ss19/
- Achten Sie darauf, in der Ausarbeitung eigenständig zu formulieren und alle verwendeten Quellen vollständig und korrekt zu zitieren.

Hinweise zum Halten des Vortrags

Im folgenden steht *Tafel* bzw. *Folie* für das verwendete Präsentationsmedium (egal ob analog oder digital).

- Schreiben Sie lesbar und lassen Sie Ihren Zuhörern genug Zeit zum Lesen. Vermeiden Sie es unbedingt, das gerade Geschriebene sofort wieder hinter einer anderen Tafel verschwinden zu lassen, wegzuwischen, oder zu schnell auf die nächste Folie umzuschalten. Planen Sie Ihr Tafelbild bzw. Ihre Folien.
- Schreiben Sie alle Definitionen an. Machen Sie bei allen Sätzen klar, was die genauen Voraussetzungen sind. Die logische Struktur sollte immer aus dem Tafelbild erkennbar sein.
- Versuchen Sie, Definitionen und Sätze anschaulich bzw. durch Anwendungsbeispiele zu motivieren. Oft können im Vortrag auch komplizierte Rechnungen durch geeignete geometrische Argumente ersetzt werden.
- Alle eingeführten Begriffe sollten durch Beispiele illustriert werden.
- Sprechen Sie laut und deutlich.

- Versuchen Sie, Ihre Zuhörer für Ihren Vortrag zu interessieren und beziehen Sie Ihr Publikum mit ein. Eine Frage an das Publikum gibt diesem Zeit nachzudenken, selbst wenn niemand die Antwort weiß.
- Versetzen Sie sich in Ihr Publikum hinein. Könnten Sie Ihrem Vortrag folgen, auch wenn Sie sich nicht vorher ausführlich mit dem Thema beschäftigt hätten?
- Haben Sie keine Angst vor Fragen des Publikums – freuen Sie sich lieber über das Interesse! Zwischenfragen der Zuhörer helfen Ihnen auch einzuschätzen, wie gut das Publikum folgen kann und welche Dinge Sie etwas genauer erklären sollten.

References

- [1] J. Avigad, L. de Moura, S. Kong. *Theorem Proving in Lean*, Release 3.23.0, https://leanprover.github.io/theorem_proving_in_lean/, 2021.
- [2] A. Beutelspacher. *Das ist o.B.d.A. trivial!*, neunte Auflage, Vieweg+Teubner, 2009.
- [3] A. Dold. *Lectures on Algebraic Topology*, Springer, 1980.
- [4] R. Ghrist. *Elementary Applied Topology*, ed. 1.0, Createspace, 2014. <https://www2.math.upenn.edu/~ghrist/notes.html>
- [5] V. de Silva and R. Ghrist. Coordinate-free coverage in sensor networks with controlled boundaries, *Intl. J. Robotics Research*, 25(12), 1205–1222, 2006.
- [6] D. Kozlov. *Combinatorial Algebraic Topology*, Algorithms and Computation in Mathematics, 21. Springer, 2008.
- [7] Lean community. Learning Lean, <https://leanprover-community.github.io/learn.html>
- [8] Lean community. Get started with Lean, https://leanprover-community.github.io/get_started.html
- [9] Lean community. Lean web editor, <https://leanprover-community.github.io/lean-web-editor/>
- [10] F. Mittelbach, M. Goossens, J. Braams, D. Carlisle, C. Rowley. *The L^AT_EX Companion*, zweite Auflage, Addison-Wesley, 2004.
- [11] J. R. Munkres. *Elements of algebraic topology*, Addison-Wesley, 1984.
- [12] T. Tantau. *The TikZ and PGF Packages*, <http://www.ctan.org/tex-archive/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf>