Regular Lecture 11
Theses and Papers

Prof. Dr. Jasmin Blanchette

Chair of Theoretical Computer Science and Theorem Proving

Version of May 13, 2026

Bachelor's and Master's Theses

Literature Search

Work on a thesis usually starts with a literature search. Your supervisor might give you **papers to read**, or you might find papers by following citations or using online databases such as Google Scholar.

Papers can play different roles:

- Some papers form the **basis** of your work.
- Some papers help **justify** your work by demonstrating that there is a **research gap**.
- Some papers are not directly related to your work but provide some broad **context** for it.

Literature Reading Techniques

Not all papers need to be read in full. To work efficiently, first read a paper's **abstract**, **introduction**, and **conclusion**, then decide whether to continue reading—or skimming—its core sections.

Approach papers you read with specific **questions**.
Look for the answers while reading.

Read **critically**. Question the statements made in papers, even if these are published at prestigious venues.

The Target Audience

You should write your thesis primarily for **fellow students**—not for your supervisors.

Thus, you need to explain the **fundamental concepts** behind your work that are not part of the curriculum. For example, if your project was to develop a compiler, you should briefly define and explain the different compiler phases.

The Title

As a rule, the title of a thesis should be **informative** rather than catchy.
It should capture your contribution precisely without being too long—e.g.:

Benchmarking of Quantum Generative AI Models
Strategies and Optimization Approaches for Exchanging Clauses in SAT Solvers
Haskell for Natural Language Processing: A Comprehensive Survey

The Abstract

The abstract of a thesis might be up to **half a page** long.
It usually follows the model "**context**, **main idea**, **contributions**."

The abstract should be readable by your **target audience**.
Thus, avoid abbreviations, and beware of jargon that fellow students might not know.
The best abstracts are concrete.

The abstract should not contain citations.
If necessary, refer to sources by naming the authors.

The Acknowledgment

Most theses include an acknowledgment at the beginning.
This is where you **thank** your supervisors followed by anyone else who helped you with your research project or your thesis
(e.g., friends or fellow students who read your text).

The Table of Contents

In computer science, tables of contents tend to have a **hybrid** structure:

- The outer chapters or sections have conventional titles—e.g., *Introduction*, *Preliminaries*, *Discussion*, *Conclusion*.
- The core chapters or sections have custom titles—e.g., *Design of the Garbage Collector*.

A bachelor's thesis usually consists of **sections**.
A master's thesis may consist of **chapters** or **sections**.
Chapters may contain multiple sections,
and sections may contain multiple subsections.

The Table of Contents

Should the **related work** be discussed up front or near the end?

Two approaches:

1. Introduction
2. Related Work
3. Preliminaries

   Core chapters or sections

n. Conclusion

1. Introduction
2. Preliminaries

   Core chapters or sections

$n-1$. Related Work

n. Conclusion

With the second approach, some related work might need to be discussed earlier, if the thesis builds on it.

The Table of Contents

Why place the related work early?

- The related work is **coherently located** at the thesis's beginning.
  instead of being distributed over the thesis.
- You can **refer** to the related work from the thesis's core.
- You can use the related work to identify a **research gap** or **open question** motivating your own work.

Why place the related work late?

- You can **focus first on your own work**.
- You can discuss the related work **in more depth** with respect to your own work.

Example of a Table of Contents

The Introduction

In the introduction, clearly state what the **purpose** or **use case** of your work is.
Be clear about what was done by others and what your contribution is.

Ways to achieve this:

- **State of the art vs. contribution**: You start by explaining the state of the art in your field. Do not hesitate to use the term *state of the art* to make this completely unambiguous for the reader. You then summarize your contribution. Do not hesitate to write *contribution* to make this clear.
- **Problem vs. solution**: You start by presenting a problem. You then present your solution to the problem. Do not hesitate to use the terms *problem* and *solution* to make things clear.

You can combine either approach with an **opening hook**.

The Introduction

When explaining the motivation for your work, focus on the **science**, not the metascience. Instead of writing

> *In recent years, an increasing number of computer scientists have turned their attention to the problem of explainable artificial intelligence.*

or

> *Explainable artificial intelligence is an important research topic.*

explain why the research is important, and let the facts speak for themselves.

If you have **research questions**, ask them in the introduction,
then answer them in the thesis's body.

The Preliminaries or Background

Early in your thesis, you will likely have a chapter or section called *Preliminaries* or *Background*. Some authors make this distinction:

- **Preliminaries** are basic definitions, notations, and other conventions that are used in the rest of the thesis.
- **Background** is related work on which your thesis builds.

You should define or explain any **specialized terms** that might be unfamiliar to your target audience. When in doubt, briefly summarize the important concepts.
For simple definitions and theorems, one or two sentences are often sufficient.
For more complex concepts, you might want to write a full paragraph.

The Core Chapters or Sections

The thesis's core is where you can explain **what you did** at length.

You can hardly ever give too many examples. Diagrams can also be useful.
But you should also describe at least the **core parts** of your work (e.g., the main data structures and algorithms) precisely and in full generality.
Focus on the nonobvious parts of your work, and try to explain them in simple words.

Focus on the final result, and **avoid anecdotes** about how you arrived at it.
Exceptionally, in case studies, your journey might be relevant.

Some theses have a separate chapter or section titled *Discussion*, where **results are analyzed**. You can also inline such discussion throughout the thesis.

The Related Work

The related work chapter or section is where you explain how your work fits into the **scientific edifice**.

When describing other people's work, make them the **subjects** of sentences—e.g., *Jones [37] designed the first monomorphization algorithm*.

You may feel that there is no or little related work to write about.
But if you broaden the definition of "related," you will **find something**.
Your supervisors should be able to help.

The Conclusion

The conclusion typically consists of a **summary** optionally followed by a brief discussion of **future work**.

The conclusion can be short. **One** or **two pages** might suffice. To quote Paulson: "A page of ramblings about everything you might do in the next five years, or would do if you had your time again, belongs on your blog."

The conclusion should make sense to readers who have read only the **abstract** and the **introduction**.

Examples of Theses

Let us look at a bachelor's and a master's thesis:

- Matthias Kettl, "**Fault Localization for Formal Verification: An Implementation and Evaluation of Algorithms Based on Error Invariants and UNSAT-Cores**," bachelor's thesis, Ludwig-Maximilians-Universität München, 2020.
- Wanja Sajko, "**Exploring the Potential of the Filtering Variational Quantum Eigensolver**," master's thesis, Ludwig-Maximilians-Universität München, 2023.

PhD Theses

The Table of Contents

Two types of PhD thesis:

- A **cumulative** thesis consists of a long (perhaps 30-page) introduction followed by verbatim copies of (typically three or more) published papers.
- A **noncumulative** thesis, or monograph, has a more traditional structure.
  A typical table of contents:
  1. Introduction
  2. Background

     Core chapters

  n. Conclusion

  The introduction might be 5 pages long, the background 10 pages long, and the conclusion 3 pages long. The (typically three or more) core chapters each describe a single contribution and would usually be at least 15 pages long. They may be based on published papers.

Acknowledgment of Coauthorship

Often PhD research is carried out in **collaboration** with coauthors.
The precise contributions of each author must be acknowledged in the introduction, the individual chapters, or an appendix.

Examples of Theses

Let us look at a cumulative and a noncumulative PhD thesis:

- Maximilian Bernhard, **Deep Learning Methods for Image Recognition in Remote Sensing**, PhD thesis, Ludwig-Maximilians-Universität München, 2024.
- Sophia Elisabeth Grundner-Culemann, **Formal Verification of Revocation Approaches in Identity-Based Cryptography**, PhD thesis, Ludwig-Maximilians-Universität München, 2024.

Research Papers

Why Write a Paper?

A scientific contribution exists if and only if it is **documented**.

You should write a paper if you have discovered something you are **excited** about and want to share it with other researchers.

Papers are more widely read than theses, and they are more concise.
To become well known in your research area, you must get papers published.

When you apply for an academic position, your **bibliometrics** (including your number of citations and your so-called h-index) matter a lot.
The motto "**publish or perish**" accurately describes the academic world.

Types of Papers

In addition to regular research papers, the following types of papers also exist, among others:

- **survey** papers;
- **system** papers;
- **application** papers;
- **evaluation** papers;
- **position** papers.

Venues

Research papers are published at different venues.

- **Workshops** are informal, noncompetitive events.
  Minor or preliminary research is presented there.
- **Conferences** are the main publication venue in most of computer science.
- **Journals** are the venue of choice for more mature or detailed results.

Workshop and conference papers are published in **proceedings**.

Venues

The main advantage of workshops and conferences is that you can advertise your research by giving a talk at an **in-person** event.

Most workshop and conference papers must fit within about 8 to 16 pages.
Additional details should be included in an appendix or in a technical report.

Journal articles can be much longer, which makes them suitable for publishing more **detailed accounts**.
Many journal articles are extended versions of conference papers.

Choose the venue that best suits the paper's content,
or make the paper match the venue.

The Table of Contents

Structurally, papers **resemble theses**, but the acknowledgments are often at the end, after the conclusion but before the reference list.

For a workshop or conference paper, it is especially important to compress the background, by referring to other sources.

As Paulson remarked: "Finally you get to describe your actual work. Are you on page 2 or page 7? If the latter, how did you manage to use half of your page allowance before saying anything original?"

Peer Reviewing

Your paper is reviewed by at least **two** or **three researchers** who do not have a conflict of interest.

Peer reviewing is generally **anonymous**, meaning that the reviewers' identities are not revealed to you.

Additionally, with **double-blind** reviewing, your identity is not revealed to the reviewers. If you need to cite yourself, do so as if you were a third party.
Thus, I might write *According to Blanchette (2012), . . .*.

Reviews usually include a summary, an assessment, a recommendation (e.g., "weak accept"), and a confidence level (e.g., "expert"). Often, reviewers also ask you questions and make suggestions to improve your paper.

Peer Reviewing

Journals do **several rounds** of reviewing. In each round, you must submit a new version of your manuscript and reply to the reviewers' comments.

By contrast, most conferences do a single round of reviewing.

Some conferences feature a lightweight second round called **rebuttal**.

At some conferences, your paper may be accepted conditionally.
The paper is then assigned a **shepherd**, who guides you through the revision process.

Typesetting

Most computer scientists typeset their papers using **LaTeX**.
A small minority use Microsoft Word. LaTeX is often mandated by publishers.

Some publishers work directly with the LaTeX sources.
Others import the papers into some proprietary tool.
Either way, the publisher might introduce typesetting errors during copy editing.
Fortunately, you get to check **galley proofs** before publication.