



Seminar *Probabilistic Programming*

Philipp Schröder et al.

Winter Semester 2024/25

Overview

Outline

Overview

Aims of this Seminar

Important Dates

The Topics

Final Hints

Probabilistic Programming

Probabilistic programs

Probabilistic programs = classical programs + probabilistic choice + conditioning

- choice: “execute program P with probability $\frac{2}{3}$ and Q with $\frac{1}{3}$ ”
- conditioning: “observe that value of variable x is positive”
- describe posterior probability distributions over variable output values

Applications

- Randomised algorithms (e.g., randomised Quicksort)
- Computer vision (e.g., image generation)
- Security
- Biology, coding theory, cryptographic protocols, machine learning, quantum computing, reliability analysis, ...

An Example

Virus infection

```
bool alicelInfectious = true
bool bobInfected = false
while alicelInfectious {
  prob 0.1 {
    bobInfected = true
  }
  prob 0.6 {
    alicelInfectious = false
  }
}
```

- What is the probability of Bob becoming infected?
- How long is Alice likely to be infectious?

About Us: The Software Modeling and Verification Group (MOVES)

- We study, develop, and apply formal methods to software systems
- Some of our teaching activities:
 - Model Checking (Katoen)
 - Practical Course Model Checking (Quatmann)
 - Semantics and Verification of Software (Noll)
 - Probabilistic Programming (Katoen)
 - Static Program Analysis (Noll)
 - Being Vice-Rector for Teaching and Learning (Katoen)
- We do research in in all areas we have courses about!
- Some of the tools we build:
 - *Storm*: a probabilistic model checker
 - *Caesar*: a deductive verification infrastructure for probabilistic programs

Contact us for thesis topics (Bachelor/Master) or for HiWi positions (teaching/research).

Areas Covered in this Seminar

Topic areas

1. Semantics

- mathematical approaches to formally define precise meaning of programs

2. Verification

- providing correctness proofs for programs (e.g. termination)

3. Program Analysis

- equivalence refutation, inference, ... based on source code

Aims of this Seminar

Outline

Overview

Aims of this Seminar

Important Dates

The Topics

Final Hints

Aims of this Seminar

Goals

Aims of this seminar

- **Independent understanding** of a scientific topic
- Acquiring, reading and understanding **scientific literature**
 - given references sufficient in most cases
- Writing of your **own report** on this topic
 - far more than just a translation/rewording
 - usually an **“extended subset”** of original literature
 1. “subset”: present core ideas and omit too specific details (e.g., related work or optimisations)
 2. “extended”: more extensive explanations, examples, ...
 - discuss contents with supervisor!
- **Oral presentation** of your results
 - can be “proper subset” of report
 - generally: less (detailed) definitions/proofs and more examples

Aims of this Seminar

Requirements on Report

Your report

- Independent writing of a report of **12–15 pages**
- First milestone: **detailed outline and initial text**
 1. overview of structure (section headers, main definitions/theorems)
 2. and initial part of main section (one page)
- **Complete** set of references to all consulted literature
- **Correct citation** of important literature
- **Plagiarism:**
 - taking text blocks (from literature or web) without source indication causes immediate **exclusion from this seminar**
 - Do not use LLMs such as ChatGPT to write your texts!
- Font size **12pt** with “standard” page layout (L^AT_EX template is on web page)
- **Language:** German or English (English recommended)
- We expect the **correct usage** of spelling and grammar
 - ≥ 10 errors per page \implies abortion of correction

Aims of this Seminar

Requirements on Talk

Your talk

- Talk of **30 minutes**
- Available: projector and presenter
- Focus your talk on the **audience**
- **Descriptive** slides:
 - ≤ 15 lines of text
 - use (base) colors in a useful manner
 - number your slides
- **Language:** German or English
- No spelling mistakes please!
- Finish **in time**. Overtime is bad
- Ask for **questions**
- Have **backup slides** ready for expected questions
- **L^AT_EX/beamer template** will be made available on seminar web page

Important Dates

Outline

Overview

Aims of this Seminar

Important Dates

The Topics

Final Hints

Important Dates

Important Dates

Deadlines

- Tue, Oct 15, 23:59: Topic preferences due
- Mon, Nov 11, 23:59: Detailed outline due
- Mon, Dec 9, 23:59: Full report due
- Mon, Jan 13, 23:59: Presentation slides due
- Feb 3–5 (?): Seminar talks

Important

Missing a deadline causes **immediate exclusion** from the seminar

(But if you want to quit, please send us a message)

Important Dates

Selecting Your Topic

Procedure

- You can find the full list of topics and all organizational details **on our web page**.
- Indicate a preference of your topics (first, second, third) on our form.
- Return sheet here or by email (phisch@cs.rwth-aachen.de) **by Tue, Oct 15, 23:59**
- We do our best to find an adequate topic-student assignment.
 - disclaimer: no guarantee for an optimal solution
- Assignment will be published on web page Oct 16.
- Then also your **supervisor** will be indicated.
- Please give language preference (unsure \implies German).

Withdrawal

- You have up to **one week** (!) to refrain from participating in this seminar.
- Later cancellation (by you or by us) causes a **not passed** for this seminar and reduces your (three) possibilities by one.

The Topics

Outline

Overview

Aims of this Seminar

Important Dates

The Topics

Final Hints

Semantics

Semantics

1. *Paradoxes of probabilistic programming: and how to condition on events of measure zero with infinitesimal probabilities*
 - probabilistic program semantics have surprising paradoxes that need to be considered
2. *Exact Recursive Probabilistic Programming*
 - semantics of recursive probabilistic programs that allow exact inference
3. *Lilac: A Modal Separation Logic for Conditional Probability*
 - transfers separation logic (originally for heap reasoning) to reasoning about probability spaces
 - *separating conjunction* captures probabilistic independence
4. *Outcome Logic: A Unifying Foundation for Correctness and Incorrectness Reasoning*
 - logic for proving and disproving properties about programs
 - also has support for probabilistic programs
5. *A Pre-expectation Calculus for Probabilistic Sensitivity*
 - sensitivity: how much does an output change depending on a change in the input?
 - based on probabilistic weakest pre-expectations (wp) a la Dijkstra

Verification

Verification

6. *Positive Almost-Sure Termination: Complexity and Proof Rules*
 - new proof rule for termination in finite expected runtime (PAST)
7. *Strong Invariants Are Hard: On the Hardness of Strongest Polynomial Invariants for (Probabilistic) Programs*
 - focused on polynomial programs (polynomial assignments and restricted structure)
8. *Sound and Complete Proof Rules for Probabilistic Termination*
 - this is about almost-sure termination (termination with probability 1)
9. *Almost-Sure Termination by Guarded Refinement*
 - using refinement to prove almost-sure termination
10. *On Lexicographic Proof Rules for Probabilistic Termination*
 - proof rules based on lexicographic ordering (i.e. check next part only if previous part is equal)
11. *Reasoning about Grover's quantum search algorithm using probabilistic wp*
 - using probabilistic weakest pre-expectations (wp)
 - verifying a *quantum algorithm* with non-quantum reasoning

Program Analysis

Program analysis

12. *Equivalence and Similarity Refutation for Probabilistic Programs*
 - a tool for refuting equivalence and refutation
13. *Compiling Probabilistic Programs for Variable Elimination with Information Flow*
 - decomposing an inference problem into subproblems using information flow
14. *Bit Blasting Probabilistic Programs*
 - discretizing continuous distributions
 - inference done on resulting discrete problem
15. *Inference of Probabilistic Programs with Moment-Matching Gaussian Mixtures*
 - semantics using *Gaussian mixtures*
 - a tool that automatically computes properties with this semantics

Final Hints

Outline

Overview

Aims of this Seminar

Important Dates

The Topics

Final Hints

Final Hints

Some Final Hints

Hints

- Take your time to **understand** your literature.
- Start early. You cannot start on the report or the talk just two days before.
- Be **proactive**! Look for **additional** literature and information.
- Search online for **recorded presentations** of research papers.
- Discuss the content of your report with other students.
- **Contact your supervisor** and have meetings.
- **Prepare** the meetings with your supervisor.

We wish you success and look forward to an enjoyable and high-quality seminar!