Bachelor & Master Projects and Theses

Prof. Dr. Stefan Leue

Software and Systems Engineering

http://sen.uni-konstanz.de/

Winter Term 2021/22
Members

- Prof. Dr. Stefan Leue {SL}
  - Email: Stefan.Leue@uni.kn, Room: PZ 902
- Dr. Georgiana Caltais {GC}
  - Email: Georgiana.Caltais@uni.kn, Room: PZ 913
- Martin Kölbl {MK}
  - Email: Martin.Koelbl@uni.kn, Room: PZ 912
- Fabian Bauer-Marquart {FBM}
  - Email: Fabian.Marquart@uni.kn, Room: PZ 912
Projects at our Chair

- **Safety Analysis, Fault Localization and Causality**
  - causality checking, Functional Safety of Automotive Systems

- **Analysis and Automated Repair of Timed Traces**
  - synthesis of repairs using SMT technology

- **QuantUM and QuantUM+**
  - Model Based System Engineering, implementation of Causality Checking

- **Architectures for Automotive Systems**
  - HW/SW Architectures for Autonomous Driving

- **Formal Verification for Machine Learning**
  - quality assurance for ML-based systems

- **Formal Verification for Quantum Computing**
  - Automated quantum program correctness proofs

- **Computational Methods in Systems Biology**
  - formal explanatory modeling of collective behavior

- **Legal Tech**
  - logical modeling and analysis of legal artefacts
Projects and Theses at the Chair

♦ Our Objectives
  ‣ projects and theses close to ongoing research projects
  ‣ links to practical and relevant applications
  ‣ completion of project and theses within defined time limits (examination regulations / Prüfungsordnung)

♦ What We Offer
  ‣ close and individual supervision
  ‣ regular meetings and guidance
  ‣ if possible and applicable, supervision in collaboration with industrial partners
Our Expectations

- **project** is typically a literature survey, problem statement or similar
  - leads to definition of thesis topic (not mandatory, but recommended)
  - project report: approx. 10-20 p.

- **thesis**
  - requires some own contribution
    - **Bachelor**: problem solution idea, critical literature survey, innovative case study, ...
    - **Master**: own problem solution concept, evolving an existing approach, algorithmic concept and implementation, revealing comparison with other approaches, ...
Scope and Duration of Projects/Theses

♦ Project (Bachelor and Master)
  ‣ 1 semester
  ‣ 9 ECTS (270h work)

♦ Thesis (Bachelor)
  ‣ 3 months (1/2 Semester)
  ‣ 12 ECTS (Thesis) + 3 ECTS (Colloquium) = 15 ECTS (450h work)

♦ Thesis (Master)
  ‣ 6 Months (1 Semester)
  ‣ 30 (Thesis + Colloquium) ECTS (900h work)
Project Report / Thesis Structure

♦ Typical Generic Structure:

1. **Introduction**
   - motivation of work, state of the art, related work, contributions

2. **Preliminaries**
   - which facts / concepts / definitions / algorithms / approaches / methods does this work rely on (“standing on the shoulders of giants”)
   - i.e., any technical information that is needed but not developed in the course of this report / thesis

3. **Approach**
   - technical contribution of the thesis (concepts / definitions / algorithms / approaches / methods etc.)

4. **Implementation**
   - software that has been implemented

5. **Evaluation**
   - case studies, experiments, quantitative and qualitative assessment, etc.

6. **Conclusion**
   - what has been accomplished
   - future research directions

7. **Bibliography**
Formal Requirements

♦ Before you start your work
    submit written proposal (≈ 1-2 pages) to sen@uni-konstanz.de containing
    – the topic you want to choose
    – how well you match the prerequisites
    – schedule for the project / thesis
      • what will be achieved at which point in time
        * requires a careful break-down of the project / thesis topic into subgoals
      • when will the project / thesis be officially registered
        (proposal for this term ideally submitted by November 18, 2021)

♦ During your preparation of the project work / thesis
    regular consultation with your supervisor
    – approx. every 4 weeks
Deliverables

- project report to the supervisor
- thesis
  - must be submitted to the examination office
  - in parallel: electronic copy (pdf) to supervisor
- any models / code / data / binaries you created for the project
  - include in DVD attached to the thesis
  - in parallel: electronic copy to supervisor
Projects and Theses

♦ Topic Areas
  ‣ Topic I: System Safety and Analysis
  ‣ Topic II: Formal Modeling and Analysis
  ‣ Topic III: Formal Verification of Machine Learning
  ‣ Topic IV: Quantum Computing and Formal Verification
  ‣ Topic V: Applications
1. **Linking QuantUM to LTSmin / PINS [B]**
   - make trace computation more memory-efficient
   - linking QuantUM to LTSmin / PINS

2. **Causality Checking for Programs [M]**
   - What causes a program to crash?
   - prototype a tool that finds failing variable assignments in programs (symbolically)

3. **Causality Checking in HyperLTL [M]**
   - HyperLTL can specify properties over program traces
   - enhance causality checking to repair programs according to HyperLTL specifications

4. **Causal Explanations of Accidents [M]**
   - relate QuantUM-based causal analysis to accident behavior of F1/10 model car

5. **Case Study on Fail-Operation Mode [B]**
   - implement fail-operational automated driving on F1/10
1. **Causality in Hybrid Systems [M]**
   - hybrid systems have continuous and discrete behavior
   - design an algorithm that computes causes of failure

2. **Repair for Parametric Timed Automata (PTAs) [B/M]**
   - PTAs are used to simulate and verify critical real-time systems
   - develop a repair procedure for PTAs

3. **Multiple Constraint Relaxation for Timed Systems [M]**
   - use multi-objective optimization to optimally relax multiple timing constraints

4. **SysML and Papyrus Interface to Promela / PINS [B]**
   - develop and implement a conversion method between modeling languages

5. **Run-Time Causality Checking [M]**
   - detect and analyze error causes during a system run-time
1. **Cause Identification of Neural Network Errors**
   - Can errors in a neural network be explained?
   - detect samples in the training data that are responsible for specification violations

2. **Counterfactual Causality in Neural Networks**
   - neural networks cannot recognize new data they have not been specifically trained for
   - for an unexpected prediction, identify and explain the causality for this prediction

3. **Verification of a Self-Driving AI**
   - DeepDrive (https://deepdrive.io) and F1tenth (https://f1tenth.org) simulate self-driving cars
   - identify safety constraints and apply them to the AI training procedure
1. Verification & Repair of Quantum Programs
   - derive an efficient SMT encoding for quantum programs
   - use over-approximation to enhance performance

2. Quantum AI Verification
   - Quantum convolutional neural networks (QCNNs) can classify classical image and quantum data
   - define meaningful safety specifications for quantum classifiers
   - implement in PyTorch and Qiskit

3. A Quantum Approach to Software Verification
   - Quantum computers can solve satisfiability problems more efficiently than classical ones
   - identify benchmarks for bounded model checking (BMC)
   - design a quantum BMC algorithm (based on SAT solving), and evaluate its complexity
1. **Modeling Collective Behavior [M]**
   - formally model emergent behavior of biological collectives
   - cooperate with the Jordan Lab @Uni KN (biology / Max-Planck)
   - select methods and tools to perform a case study on a school of fish

2. **LegalTech: Logical Analysis of Sales Contracts [B/M]**
   - sales contracts need to be well-defined and self-consistent
   - develop an algorithm that detects inconsistencies in sales contracts

3. **Explaining Faults with Machine Learning [B/M]**
   - Model Checking algorithms give way too many counter-examples in case of errors
   - analyze these counter-examples using machine learning and detect patterns
For BA Projects and Theses, the Following Dates Apply

<table>
<thead>
<tr>
<th>Projekt</th>
<th>Anmeldung</th>
<th>Abgabe bis</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15.10. – 15.11.</td>
<td>15.01.</td>
</tr>
<tr>
<td></td>
<td>15.01. – 15.02.</td>
<td>15.04.</td>
</tr>
<tr>
<td></td>
<td>01.04. – 01.05.</td>
<td>30.06.</td>
</tr>
<tr>
<td></td>
<td>01.07. – 01.08.</td>
<td>30.09.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Abschlussarbeit</th>
<th>Anmeldung</th>
<th>Bearbeitungsbeginn</th>
<th>Abgabe bis*</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>01.02. – 15.02.</td>
<td>01.03.</td>
<td>01.06.</td>
</tr>
<tr>
<td></td>
<td>01.05. – 15.05.</td>
<td>01.06.</td>
<td>01.09.</td>
</tr>
<tr>
<td></td>
<td>15.07. – 01.08.</td>
<td>15.08.</td>
<td>15.11.</td>
</tr>
<tr>
<td></td>
<td>15.10. – 02.11.</td>
<td>15.11.</td>
<td>01.03.</td>
</tr>
</tbody>
</table>

* ungefähre Angabe; der genaue Zeitpunkt wird vom ZPA festgelegt
Important

♦ Own Ideas Welcome!

› if you have own ideas
  – topics not included in our catalog
  – modifications of proposed topics

please talk to us!

• topic finding is an iterative, deliberative process!
... either one of us at any time!

- Prof. Dr. Stefan Leue
  - Email: Stefan.Leue@uni.kn
- Martin Kölbl
  - Email: Martin.Koelbl@uni.kn
- Fabian Bauer-Marquart
  - Email: Fabian.Marquart@uni.kn

or: sen@uni-konstanz.de