

# Zafer Esen

✉ zafer.esen@gmail.com

🔗 <https://github.com/zafer-esen/>

Webpage: <https://zafer-esen.github.io>

## Education

- 2019 – 2025.06 PhD in Computer Science, Uppsala University, Sweden  
Supervisors: Philipp Rümmer (main), Tjark Weber, Wang Yi
- 2017 – 2019 MSc in Embedded Systems, Uppsala University, Sweden
- 2005 – 2009 BSc in Electronics Engineering, Ankara University, Turkey

## Job Experience

- 2024.05 – 2024.08 Applied Science Intern, Amazon, Boston, United States  
*I worked in the Dafny team that is part of the Automated Reasoning group at AWS, where my job was to work on the development of a new program equivalence checker.*
- 2023.06 – 2023.09 Applied Science Intern, Amazon, Boston, United States  
*I worked in the Dafny team that is part of the Automated Reasoning group at AWS, where my job was to work on methods to reduce proof brittleness in Dafny.*
- 2011 – 2017 Control Systems Design Engineer, Roketsan, Turkey  
*(I mainly worked as a control and embedded systems design engineer. My main duties included designing and testing control systems in Simulink, as well as the hardware and software that implement such systems.)*

## Relevant Skills

- SMT I work with SMT on a regular basis as part of my research. I am one of the contributors to the PRINCESS SMT solver, and we have a recent paper proposing a new SMT-LIB theory of heaps (that we also implemented in PRINCESS).
- Model checking I am familiar with Horn-based model checking and I am one of the developers of the Horn-based C model-checker TRICERA.
- Deductive Verification I use Dafny in a course I have been teaching for the past few years (1DT034 Programming Theory). In the course I also teach the basics of propositional and predicate logic, as well as program proofs using Hoare logic and weakest pre-conditions calculus. I was part of the core team that develops Dafny during my internships, and have contributed to both Boogie and Dafny.
- Programming Skills I have experience with several programming languages including Ada, C, C++, C#, Dafny, Java, Scala. I use Scala on a daily basis.

## Service

- PC member HCVS (25)
- Peer reviews (as subreviewer) IJCAR (20, 22, 24), TACAS (20, 23, 24), CONCUR (20), FSEN (21), NWPT (21), CAV (23), FMCAD (22, 23, 24), FM (24), FRODOS (25)
- Artifact evaluations  
(as artifact evaluation committee member) CAV (21, 23), TACAS (23, 24), VMCAI (24, 25)

## Talks

- 2022.08 “ELDARICA and TRICERA: Towards an Open Verification Framework”, Zafer Esen. DSV 2022 <https://smackers.github.io/democratizing-software-verification-workshop-2022/>
- 2021.04 “Towards Automatic Verification of Unsafe Rust with Constrained Horn Solvers”, Zafer Esen, Philipp Rümmer, Amanda Stjerna. RW 2021, <https://sites.google.com/view/rustverify2021>

## Tools

- |                         |                |   |
|-------------------------|----------------|---|
| TRICERA C Model Checker | (co-developer) | <a href="https://github.com/uuverifiers/tricera">https://github.com/uuverifiers/tricera</a>   |
| ELDARICA Horn Solver    | (contributor)  | <a href="https://github.com/uuverifiers/eldarica">https://github.com/uuverifiers/eldarica</a> |
| PRINCESS SMT Solver     | (contributor)  | <a href="https://github.com/uuverifiers/princess">https://github.com/uuverifiers/princess</a> |

## Publications

-  2025 “Transformations for Verifying Programs with Heap-Allocated Data Structures”, Zafer Esen, PhD Thesis, <https://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-554456>
-  2025 “Sound and Complete Invariant-Based Heap Encodings” (under submission, preprint at <https://arxiv.org/abs/2504.15844>)
-  2025 “Finding Universally Quantified Heap Invariants by Horn Clause Transformations”, Zafer Esen, Philipp Rümmer, Tjark Weber, FSEN 2025
-  2025 “Arithmetizing Shape Analysis”, Sebastian Wolff, Ekanshdeep Gupta, Zafer Esen, Hossein Hojjat, Philipp Rümmer, Thomas Wies, CAV 2025 (to appear, preprint at <https://arxiv.org/pdf/2408.09037>)
-  2024 “An Exercise in Mind Reading: Automatic Contract Inference for Frama-C”, Jesper Amilon, Zafer Esen, Dilian Gurov, Christian Lidström, Philipp Rümmer, book chapter in “Guide to Software Verification with Frama-C. Core Components, Usages, and Applications”, Springer Nature.
-  2023 “Automatic Program Instrumentation for Automatic Verification”, Jesper Amilon, Zafer Esen, Dilian Gurov, Christian Lidström, Philipp Rümmer, CAV 2023 (**Distinguished Paper Award**)
-  2022 “TRICERA: Verifying C Programs Using the Theory of Heaps”, Zafer Esen, Philipp Rümmer. FMCAD 2022
-  2022 “An SMT-LIB Theory of Heaps”, Zafer Esen, Philipp Rümmer. SMT 2022
-  2021 “A Theory of Heap for Constrained Horn Clauses”, Zafer Esen, Philipp Rümmer, 32nd Nordic Workshop on Programming Theory NWPT 2021, <http://icetcs.ru.is/nwpt21/abstracts/paper26.pdf>
-  2021 “A Theory of Heap for Constrained Horn Clauses (Extended Technical Report)”, Zafer Esen, Philipp Rümmer, CoRR abs/2104.04224, <https://arxiv.org/abs/2104.04224>
-  2021 “Reasoning in the Theory of Heap: Satisfiability and Interpolation”, Zafer Esen, Philipp Rümmer, LOPSTR 2020: 173-191 (*invited paper for Philipp Rümmer*), [https://doi.org/10.1007/978-3-030-68446-4\\_9](https://doi.org/10.1007/978-3-030-68446-4_9)
-  2020 “Towards an SMT-LIB Theory of Heap”, Zafer Esen, Philipp Rümmer, EPTCS, volume 320, pages 159-162, <https://doi.org/10.4204/EPTCS.320>
-  2019 “Extension of the ELDARICA C model checker with heap memory”, Zafer Esen, MSc Thesis, <http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-397812>

Legend:  Abstract and talk,  Book Chapter,  Paper,  Technical Report,  Thesis.

## Supervised Students

2024.08 - 2025.06	Daniel Wallgren, MSc. thesis, “Evaluating Symbolic Execution Strategies on Constrained Horn Clauses”
2024.03 - 2024.06	Kalle Nordgren, BSc. thesis, “Graph-Based Clausification for ELDARICA”
2024.03 - 2024.06	Isak Drevstad, BSc. thesis, “Parallel / Distributed State-Space Exploration in an Infinite-State Model Checker”
2024.03 - 2024.06	Fredrik Jäderblom, BSc. thesis, “Implementing and Evaluating FLATA Compatibility with Modern SMT Solvers”
2023.03 - 2023.06	Danyal Mirza, BSc. thesis, “Adding Support for Floating-Point Arithmetic to TRICERA”
2023.01 - 2023.03	Axel Bergström, BSc. thesis, “Adding Basic Support for Function Pointers in TRICERA”
2021.09 - 2022.03	Pontus Ernstedt, MSc. thesis, “Contract-Based Verification in TRICERA”

## Teaching

2024	Programming Theory (1DT034)	Teacher
2023	Programming Theory (1DT034)	Teacher
2022	Programming Theory (1DT034)	Teacher
2021	Programming Theory (1DT034)	Teacher
2020	Real Time Systems (1DT004)	Teaching assistant
2020	Programming Theory (1DT034)	Teaching assistant
2020	Accelerating Systems with Programmable Logic Components (1DT109)	Teaching assistant
2019	Real Time Systems (1DT009)	Teaching assistant
2019	Programming Theory (1DT034)	Teaching assistant

*(All courses were taught at Uppsala University.)*