

Amirmohammad Nazari

3737 Watt Way, PHE 101, Los Angeles, CA 90089
nazaria@usc.edu
+1 (310) 621-0676

RESEARCH INTERESTS

- Automated Reasoning
- Formal Methods
- Formal Verification
- Programming Languages
- Program Synthesis
- Program Analysis

EDUCATION

- **University of Southern California**, Los Angeles, California, USA
Ph.D., Computer Science
May 2021 - Ongoing
Thesis : To be announced
Advisor : Mukund Raghothaman
- **Amirkabir University of Technology**(Tehran Polytechnic), Tehran, Iran
B.Sc., Computer Engineering
September 2016 - September 2020
Thesis : Modeling and design and implementation of transparent value-sensitive run-time monitoring for information flow policies
Advisor: Mehran S. Fallah

HONORS AND AWARDS

- Awarded **Direct Admission** to graduate school (M.Sc.) of Computer Engineering, Amirkabir University of Technology, without taking the national entrance exam for graduate schools as a reward of academic records and achievements, Tehran, Iran, 2019.
- Ranked **five** regarding cumulative GPA among all students of Computer Engineering (more than 150 students), Amirkabir University of Technology, Tehran, Iran, 2019.
- Achieved top **1.5%** place among all applicants for the University Entrance Nationwide Exam (Approximately 163000 applicants) in Math. and Eng., Iran, 2016.

PUBLICATIONS

- Amirmohammad Nazari, Yifei Huang, Roopsha Samanta, Arjun Radhakrishna, and Mukund Raghothaman. Explainable Program Synthesis by Localizing Specifications. In submission. 2022.

RESEARCH EXPERIENCES

- **Explainable synthesis by localizing specifications**
The classical formulation of the program synthesis problem is to find a program that meets a correctness specification given as a logical formula. When the program synthesis is successful, there is a guarantee that the implementation satisfies the specification. Unfortunately, program synthesis engines are typically monolithic algorithms, and obscure the correspondence between the specification, implementation and user intent. In contrast, humans often include comments in their code to guide future developers towards the purpose and design of different parts of the codebase. In this paper, we introduce *sub-specifications* as a mechanism to augment the output of program synthesizers with explanatory notes of this form. We investigate their theoretical properties,

develop algorithms to construct and validate sub-specifications, and perform an experimental evaluation in which we measure their running times and effectiveness in simplification. We submitted the paper for POPL 2023.

- **Recursive Program Synthesis by Saturated Enumeration**

Recursive program synthesis is the task to construct a recursive program that provably satisfies a given high-level formal specification. There has been a growing interest in recursive program synthesis. In spite of advances achieved over the past decade, there remain some challenges that are in need of further research. A more recent technique known as equality saturation repurposes e-graphs to implement state-of-the-art, rewrite-driven compiler optimizations and program synthesizers. In this paper, we use the idea of equality saturation to introduce a new technique for recursive program synthesis. The research project still is in progress under the supervision of Prof. Mukund Raghothaman.

- **Implementation of transparent value-sensitive run-time monitoring for information flow policies - B.Sc. Project**

My implementation and design is based on an article named "On transparent value-sensitive run-time monitoring for information flow policies" written by Fatemeh Iranimehr and Mehran S. Fallah. Information flow policies demand that confidential information should not flow to public objects which unauthorized entities can observe. Purely dynamic monitors and hybrid monitors enforce security policies by observing and transforming the executions of target programs. I implemented purely dynamic monitors and hybrid monitors for JavaScript. I inlined monitors inside JavaScript programs so that confidential information do not flow to public objects which unauthorized entities can observe. The B.Sc. project were under the supervision of Prof. Mehran S. Fallah.

TEACHING and MENTORING EXPERIENCES

- **Mentor, SHINE Program,** Summer 2022
Computer Science Department, University of Southern California
Under supervision of Prof. Mukund Raghothaman
- **Mentor, CURVE Program,** Spring 2022
Computer Science Department, University of Southern California
Under supervision of Prof. Mukund Raghothaman
- **Mentor, CURVE Program,** Fall 2021
Computer Science Department, University of Southern California
Under supervision of Prof. Mukund Raghothaman
- **Teaching Assistant, Programming Languages,** Fall 2020
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Mehran S. Fallah
- **Teaching Assistant, Computer Architecture,** Spring 2020
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Hamed Farbeh
- **Teaching Assistant, Programming Languages,** Spring 2020
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Mehran S. Fallah

- **Teaching Assistant, Programming Languages,** Fall 2019
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Mehran S. Fallah
- **Teaching Assistant, Principles of Compiler Design,** Fall 2019
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Mohammadreza Razzazi
- **Teaching Assistant, Computer Architecture,** Spring 2019
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Hamed Farbeh
- **Teaching Assistant, Discrete Mathematics,** Spring 2019
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Mehran S. Fallah
- **Teaching Assistant, Data Structures And Algorithms,** Fall 2018
Computer Engineering Department, Amirkabir University of Technology
Under supervision of Prof. Alireza Bagheri

REFERENCES

- **Mukund Raghothaman, Assistant Professor**
Computer Science Department, University of Southern California
Email: raghotha@usc.edu
- **Roopsha Samanta, Assistant Professor**
Department of Computer Science, Purdue University
Email: roopsha@purdue.edu
- **Arjun Radhakrishna, Researcher**
PROSE team, Microsoft
Email: arradha@microsoft.com