

Mahbod Afarin

PHD CANDIDATE, GRADUATE RESEARCH ASSISTANT, UNIVERSITY OF CALIFORNIA RIVERSIDE

RESEARCH INTERESTS	<ul style="list-style-type: none"> • Graph Processing Algorithms & Accelerators • GPU Architecture & Programming 	<ul style="list-style-type: none"> • Computer Architecture • Compiler Design
SKILLS	<ul style="list-style-type: none"> • Programming Languages: C/C++, Python, CUDA, OpenMP, OpenCL, MATLAB. • Compiler: Working knowledge of LLVM based on class projects as TA. • Hardware Design: VHDL, Verilog HDL, SystemC, Xilinx ISE, Altera Quartus, Celoxica Agility Compiler, Synopsys Design Compiler. • Simulation Tools: Multi2Sim, GPGPU-Sim, Mentor Graphics Modelsim, HSPICE, PSPICE, IC Encounter, HSIM, Cadence SoC Encounter, The Structural Simulation Toolkit. 	
PUBLICATIONS	<ul style="list-style-type: none"> • Xiaolin Jiang, Mahbod Afarin, Zhijia Zhao, Nael Abu-Ghazaleh, Rajiv Gupta, “Core Graph: Exploiting Edge Centrality to Speedup the Evaluation of Iterative Graph Queries,” <i>2024 Proceedings of the Nineteen European Conference on Computer Systems (EuroSys’24)</i> (Contributed Equally with the First Author). • Chao Gao, Mahbod Afarin, Shafiur Rahman, Nael Abu-Ghazaleh, Rajiv Gupta, “MEGA Evolving Graph Accelerator,” <i>2023 56th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO’23)</i> (Contributed Equally with the First Author). • Mahbod Afarin, Chao Gao, Shafiur Rahman, Nael Abu-Ghazaleh, Rajiv Gupta, “CommonGraph: Graph Analytics on Evolving Data (Abstract),” <i>In Proceedings of the 2023 ACM Workshop on Highlights of Parallel Computing (HOPC’23)</i>. • Mahbod Afarin, Chao Gao, Shafiur Rahman, Nael Abu-Ghazaleh, Rajiv Gupta, “CommonGraph: Graph Analytics on Evolving Data,” <i>International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS’23)</i>. • Shafiur Rahman, Mahbod Afarin, Nael Abu-Ghazaleh, Rajiv Gupta, “JetStream: Graph Analytics on Streaming Data with Event-Driven Hardware Accelerator,” <i>2021 54th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO’21)</i>. • Abbas Mazloumi, Mahbod Afarin, Rajiv Gupta, “Expressway: Prioritizing Edges for Distributed Evaluation of Graph Queries,” <i>IEEE Tenth International Workshop on High Performance Big Graph Data Management, Analysis, and Mining, colocated with IEEE BigData</i>, pages 1-10, Sorrento, Italy, December 2023. 	
AWARDS & ACHIEVEMENTS	<ul style="list-style-type: none"> • Won UCR GSA Travel Grant Award at University of California, Riverside, 2023. • Won Dean’s Distinguished Fellowship Award at University of California, Riverside, 2019. • Ranked 7th in terms of total GPA among 83 Computer Engineering students in Sharif University of Technology (Top 8%), 2018. • Admitted as an Exceptional Talent at Sharif University of Technology for M.Sc, 2015. • 1st Rank, Achievement of the highest GPA in B.Sc among all Computer Engineering graduated students in Shahed University, 2015. 	
RESEARCH EXPERIENCE	<ul style="list-style-type: none"> • Graduate research Assistant: Member of the GRaph Analytics with Scalability & Performance (GRASP) research group under supervision of Prof. Rajiv Gupta & Prof. Nael Abu-Ghazaleh (<i>Jan’ 20 - Present</i>). • Graduate research Assistant: Member of the RIverside Programming Language & Software Engineering (RIPLE) research group under supervision of Prof. Rajiv Gupta & Prof. Nael Abu-Ghazaleh (<i>Jan’ 20 - Present</i>). • Member of Very Large Scale Integration Laboratory (VLSI-Lab) under supervision of Prof. Shaahin Hessabi (<i>Dec’ 15 - Jan’ 18</i>). 	

EDUCATION

- **Doctor of Philosophy (Ph.D.)**, Computer Science, University of California Riverside, California, USA. *Jan' 20 - Dec' 24 (Expected)*
 - **Thesis:** *"Hardware-Software Approaches for Accelerating Graph Processing Workloads"*
 - **Advisors:** [Professor Rajiv Gupta](#) & [Professor Nael Abu-Ghazaleh](#)
 - **GPA:** 3.86/4
- **Master of Science (M.Sc.)**, Computer Engineering (Computer System Architecture), Sharif University of Technology, Tehran, Iran. *Sep' 15 - Jan' 18*
 - **Thesis:** *"Improving Manufacturing Yield and Life Cycle of Special Purpose SIMT Processors for Inexact Computing"*
 - **Advisors:** [Professor Shaahin Hessabi](#)
 - **Thesis Grade:** Excellent
 - **GPA:** 4/4 (19.03/20) (Ranked **7th** in terms of total GPA among 83 Computer Engineering students – Top **8%**)
- **Bachelor of Science (B.Sc.)**, Computer Engineering (Computer System Architecture), Shahed University, Tehran, Iran. *Sep' 11 - Jun' 15*
 - **Thesis:** *"Comparing different types of software for designing with SystemC and implementing Mano processor with SystemC"*
 - **Advisors:** [Professor Naser Mohammadzadeh](#)
 - **Thesis Grade:** Excellent
 - **GPA:** 3.63/4 (17.53/20) (Ranked **1st** among all Computer Engineering graduate students)
 - **GPA (Expertise Courses):** 3.83/4 (18.03/20)
 - **Last year GPA:** 4/4 (19.41/20)
 - **Last two years GPA:** 3.96/4 (18.88/20)

SELECTED RESEARCH PROJECTS

- **JetStream (MICRO'21): Event-Driven Hardware Accelerator for Streaming Graph Analytics**
UC Riverside, Riverside, CA *Jun 2020 – Apr 2021*
 - **First Streaming Graph Accelerator:** Developed **JetStream**, the first accelerator to support operations on streaming graphs (or dynamic graphs).
 - **New Asynchronous Streaming Algorithms:** **JetStream** subsumes the capabilities of **GraphBolt** and **KickStarter** software frameworks that allow edge deletions.
 - **Large Performance Improvements that improve with smaller batch sizes:** **JetStream** substantially outperforms both software frameworks. In addition, its advantage grows as the batch size is reduced, making it suitable for near real-time analytics.
- **MEGA (MICRO'23): Evolving Graph Accelerator**
UC Riverside, Riverside, CA *Jun 2022 – May 2023*
 - **First Evolving Graph Accelerator:** **MEGA** MEGA, the first accelerator for evolving graph workloads, provides support for multiple snapshots executing simultaneously.
 - **Batch-Oriented Execution:** Batch-oriented execution exploits the similarity of the graph across snapshots to reuse similar edge-fetches and minimize redundant execution of batches.
 - **Batch Pipelining:** We explore optimizations to the workflow to improve concurrency such as allowing multiple concurrent batches, and using pipelining across batches.
 - **Event-Driven Execution:** We develop an event-driven datapath to support the overall execution flow.
- **CommonGraph (ASPLOS'23): Graph Analytics on Evolving Data**
UC Riverside, Riverside, CA *Aug 2021 – Jun 2022*

- **Converting Expensive Deletions to Additions:** Developed a new approach to evolving graphs analysis that avoids processing of deletions by converting them to additions to the CommonGraph. CommonGraph facilitates work sharing and parallelism across snapshots.
- **Triangular Grid Data Structure for Direct-Hop and Work-Sharing Algorithm:**
 - * Developed a new structure called Triangular Grid that exposes work-sharing opportunities across snapshots. A Steiner Tree formulation finds a solution that maximizes work-sharing.
 - * Designed a graph representation that avoids the need to mutate graphs and enables reuse of edges by snapshots that share them.
 - * Implemented the CommonGraph that exploits the above ideas and delivers considerable speedups over KickStarter.
- **Core Graph (EuroSys’24): Exploiting Edge Centrality to Speedup the Evaluation of Iterative Graph Queries**
UC Riverside, Riverside, CA *May 2021 – May 2023*
 - **Core Graph Identification and Exploitation:**
 - * Our study of multiple kinds of graph queries on irregular graphs shows that the solution of a query is determined by a small fraction of total edges, i.e. critical edges. Also, many edges recur frequently across critical edge sets of different queries.
 - * Present algorithms for finding a core graph by solving a small set of queries to identify most non-zero centrality edges.
 - * Exploit core graph and present a new optimization that improves the efficiency of the 2phase evaluation while producing 100% precise results.
 - **Experimental Results:**
 - * For the 2.586 billion Friendster graph, across six kinds of queries, our approach yielded core graphs containing 5.42% to 10.45% edges and precise results for 97.1–99.9% vertices.
- **Expressway (Big Graph’23): Prioritizing Edges for Distributed Evaluation of Graph Queries**
UC Riverside, Riverside, CA *Jun 2023 – Aug 2023*
 - **Precise Results Using Highways:** Our study demonstrated that we can obtain precise results for most of the vertices by using only a small subset of the edges, i.e., *highways*.
 - **Algorithm for Identifying Highways:** We introduced a novel algorithm for identifying *highways* in a given graph.
 - **Distributed System Application:** We enhanced the performance of distributed graph query evaluation through a two-step algorithm. The algorithm first runs on the graph using only the *highways*, and then it runs on the graph using all the edges.

REVIEWING FOR
CONFERENCES
& JOURNALS

- **Conferences:** PPOPP’24, MICRO’23, ICDCS’23, ACM ICS’23, ISPASS’23, ICDCS’22, ISPASS’22, CGO’20, MICRO’20, PACT’20.
- **Journals:** CAL’23, TACO’23, IEEE Transaction on Computers’23, Parallel Computing’23.

TEACHING
EXPERIENCE

- Teaching Assistant, **Compiler Design**, University of California, Riverside, Department of Computer Science & Engineering, Summer 2023, [Prof. Rajiv Gupta](#).
- Teaching Assistant, **Compiler Design**, University of California, Riverside, Department of Computer Science & Engineering, Summer 2022, [Prof. Rajiv Gupta](#).
- Teaching Assistant, **Compiler Design**, University of California, Riverside, Department of Computer Science & Engineering, Spring 2022, [Prof. Rajiv Gupta](#).
- Teaching Assistant, **Compiler Design**, University of California, Riverside, Department of Computer Science & Engineering, Summer 2021, [Prof. Rajiv Gupta](#).
- Teaching Assistant, **Compiler Design**, University of California, Riverside, Department of Computer Science & Engineering, Spring 2021, [Prof. Rajiv Gupta](#).

- Teaching Assistant, **System on Chip** (Graduate), Sharif University of Technology, Department of Computer Engineering, Spring 2018, Prof. Shaahin Hessabi.
- Teaching Assistant, **Testability** (Graduate), Sharif University of Technology, Department of Computer Engineering, Fall 2017, Prof. Shaahin Hessabi.
- Lab Instructor, **Logic Design Lab**, Sharif University of Technology, Department of Computer Engineering, Summer 2017, Prof. Siavash Bayat-Sarmadi.
- Teaching Assistant, **Advanced VLSI** (Graduate), Sharif University of Technology, Department of Computer Engineering, Spring 2017, Prof. Shaahin Hessabi.
- Teaching Assistant, **VLSI** (Undergraduate), Sharif University of Technology, Department of Computer Engineering, Fall 2016, Prof. Shaahin Hessabi.
- Lab Instructor, **Digital System Design Lab**, Sharif University of Technology, Department of Computer Engineering, Summer 2016, Prof. Maziar Goudarzi.
- Teaching Assistant, **VLSI Design** (Undergraduate), Shahed University, Department of Computer Engineering, Fall 2019, Prof. Naser Mohammadzadeh.
- Teaching Assistant, **Computer Architecture** (Undergraduate), Shahed University, Department of Computer Engineering, Fall 2019, Prof. Naser Mohammadzadeh.
- Teaching Assistant, **Digital Electronic** (Undergraduate), Shahed University, Department of Computer Engineering, Spring 2019, Prof. Naser Mohammadzadeh.
- Teaching Assistant, **Computer Architecture** (Undergraduate), Shahed University, Department of Computer Engineering, Spring 2019, Prof. Naser Mohammadzadeh.
- Lab Instructor, **Logic Design Lab**, Shahed University, Department of Computer Engineering, Spring 2019, Prof. Naser Mohammadzadeh.
- Lab Instructor, **Digital System Design Lab**, Shahed University, Department of Computer Engineering, Spring 2019, Prof. Naser Mohammadzadeh.

SELECTED
COURSE
PROJECTS

- Used **C++** to add a real kernel thread to *xv6* (*xv6* is an instructional OS consisting of a stripped-down version of *UNIX*) using a new system call, **Advanced Operating System** course project, Prof. Nael Abu-Ghazaleh, Spring 2021 ([Github][Overview][Report]).
- Used **C++** to implement the **lottery and stride scheduling**, and a system calls that tracks the information about the system (information includes number of the process, system calls, and used pages) in *xv6*, **Advanced Operating System** course project, Prof. Nael Abu-Ghazaleh, Spring 2021 ([Github][Overview][Report]).
- Parallelize matrix multiplication using the **work-sharing** for directive using *OpenMP* (The *OpenMP* API specification for parallel programming), **Multiprocessor Architecture and Programming** course project, Prof. Elaheh Sadredini, Winter 2022 ([Github][Overview][Report]).
- Improving the performance of the **sparse matrix-vector multiplication** using *OpenMP* *OpenMP*, **Multiprocessor Architecture and Programming** course project, Prof. Elaheh Sadredini, Winter 2022 ([Github][Overview][Report]).
- Implementation of a histogram kernel and demonstration of the usage of **atomic operations** on it using *CUDA*, **GPU Architecture and Programming** course project, Prof. Daniel Wong, Fall 2021 ([Github][Overview][Report]).
- Implementation of a **tiled matrix multiplication kernel** which can support arbitrary-sized matrices using *CUDA*, **GPU Architecture and Programming** course project, Prof. Daniel Wong, Fall 2021 ([Github][Overview][Report]).
- Implementing and optimizing the **reduction kernel** and analyzing basic architectural performance properties using *CUDA*, **GPU Architecture and Programming** course project, Prof. Daniel Wong, Fall 2021 ([Github][Overview][Report]).
- Improving the performance of Vector Addition program by finding the best *basic block* size using *CUDA*, **GPU Architecture and Programming** course project, Prof. Daniel Wong, Fall 2021 ([Github][Overview][Report]).

- Performance comparison of various GPU programming APIs (*OpenMP*, *CUDA*, *OpenCL*), [GPU Architecture and Programming](#) course project, Prof. Daniel Wong, Fall 2021.
- Implementation of the fundamental available expression analysis and **common sub-expression elimination** using C++ language, [Compiler Construction](#) course project, Prof. Rajiv Gupta, Fall 2021 ([\[Github\]](#)[\[Report\]](#)).
- Implementation of the fundamental **liveness analysis** as an *LLVM* pass using C++ language, [Compiler Construction](#) course project, Prof. Rajiv Gupta, Fall 2021 ([\[Github\]](#)[\[Report\]](#)).
- Performance optimization of a matrix multiplication program via **register and cache reuse** using C++, [High Performance Computing](#) course project, Prof. Zizhong Chen, Fall 2021 ([\[Github\]](#)[\[Overview\]](#)[\[Report\]](#)).
- High performance sequential codes for **Solving Large Linear Systems**, [High Performance Computing](#) course project, Prof. Zizhong Chen, Fall 2021 ([\[Github\]](#)[\[Overview\]](#)[\[Report\]](#)).
- Parallel Sieve of Eratosthenes for Finding All Prime Numbers using **Open Message Passing Interface** (Open MPI), [High Performance Computing](#) course project, Prof. Zizhong Chen, Fall 2021 ([\[Github\]](#)[\[Overview\]](#)[\[Report\]](#)).
- Finding influencers in a network using Data Mining Techniques and SNAP Stanford Dataset collections, [Data Mining](#) course project, Prof. Vagelis Papalexakis, Spring 2021.
- Design and implementation of a $O(m \times n)$ space, $O(m \times n \times k)$ time *SP Alignment Algorithm* for three DNA sequences using JAVA language, [Algorithmic Techniques in Computational Biology](#) course project, Prof. Tao Jiang, Spring 2021.
- Implementing the **Tomasulo Speculative Algorithm** using C++ language, [Advanced Computer Architecture](#) course project, Prof. Daniel Wong, Winter 2020 ([\[Github\]](#)[\[Report\]](#)).
- Design and implementation of a **pipeline simulator** which supports **forwarding technique** using C++ language, [Advanced Computer Architecture](#) course project, Prof. Daniel Wong, Winter 2020 ([\[Github\]](#)[\[Overview\]](#)[\[Report\]](#)).
- Design and implementation of various **branch predictors**, including simple 1-bit and 2-bit predictors, as well as correlating (m,n) predictors using C++ language, [Advanced Computer Architecture](#) course project, Prof. Daniel Wong, Winter 2020 ([\[Github\]](#)[\[Overview\]](#)[\[Report\]](#)).
- Design and implementation of a cache, with support for **direct-mapped, set-associative, and fully-associative mapping**, [Advanced Computer Architecture](#) course project, Prof. Daniel Wong, Winter 2020 ([\[Github\]](#)[\[Overview\]](#)[\[Report\]](#)).
- Design and implementation of a **Search Engine** which allows users to search related tweets in the cryptocurrency area using Hadoop and Lucene, [Information Retrieval and Web Search](#) course project, Prof. Vagelis Hristidis, Winter 2020.
- Design, Simulation, and implementation of complex ALU with exciting IP cores in *ISE Design Suite - Xilinx*, and synthesize it with *Synopsis Design Compiler*, System on Chip Design course project, Prof. Shaahin Hessabi, Fall 2017.
- **Post-layout simulation** for a complex ALU with **SoC Encounter**, System on Chip Design course project, Prof. Shaahin Hessabi, Fall 2017.
- Design, Simulation, and implementation of a serial divider at gate level and synthesize it with a scan chain, Testability course project, Prof. Shaahin Hessabi, Spring 2017.
- Implementing a software for simulating the impact of fault and delay on standard combinations logic with JAVA, Testability course project, Prof. Shaahin Hessabi, Spring 2017.
- Implementing a software for simulating ISCAS combinational circuits at gate level with JAVA. Testability course project, Prof. Shaahin Hessabi, Spring 2017.
- Replicate an article which was entitled: "Energy-aware Scheduling meth of MapReduce Jobs", Green Computing course project, Prof. Maziar Goudarzi, Spring 2017.
- Investigation of Aging Mitigation Methods in Graphic Processing Units, Advanced Topics in Dependable Computing Systems project, Prof. Seyed-Ghassem Miremadi, Spring 2017.
- Investigation of the overhead of crosstalk avoidance codes for reliable data transfer of NoCs, Fault-Tolerant Systems Design, Prof. Seyed-Ghassem Miremadi, Fall 2016.

- Designing and implementing an arbiter circuit, and determining the minimum die size for it using **SoC-Encounter**, Advanced VLSI Design, Prof. Shaahin Hessabi, Fall 2016.
- Finding the optimum W-min for XOR_XNOR circuit in order to optimize power, PDP, and delay, Advanced VLSI Design, Prof. Shaahin Hessabi, Fall 2016.
- Designing, simulating, and implementing an arbiter circuit with **Verilog**, **Modelsim**, and **Synopsis Design Compiler**, and examining the effect of optimization constraints on speed and area, Advanced VLSI Design, Prof. Shaahin Hessabi, Fall 2016.
- Designing and simulating NAND gate in RTL logic and examining the effects of pull up resistor on the output load, Advanced VLSI Design, Prof. Shaahin Hessabi, Fall 2016.
- Design and Implementation of a circuit which controls direction and speed of the stepping engine motor using keyboard and LCD screen with **μ Vision IDE–Keil** and **ARM microprocessor**, Microprocessor laboratory project, Prof. Mohammad Pooyan, Spring 2015.
- Design and Implementation of Signal Generator using digital to analog converter with **μ Vision IDE** and **ARM microprocessor**, Microprocessor laboratory project, Prof. Mohammad Pooyan, Spring 2015.
- Design and Implementation of frequency meter using timer and counter with **μ Vision IDE–Keil** and **ARM microprocessor**, Microprocessor lab, Prof. Mohammad Pooyan, Spring 2015.
- Design, Simulation, and Implementation of a Double Precision Floating Point Multiplier with **VHDL** and **ISE Design**, VHDL course project, Prof. Naser Mohammadzadeh, Spring 2014.
- Design, Simulation, and Implementation of **MIPS Processor** with **Verilog**, **ISE Design Suite**, and **Xilinx FPGA**, Computer Architecture laboratory course, Prof. Naser Mohammadzadeh, Spring 2014.
- Design and Implementation of Quoridor Game with **Prolog**, Artificial Intelligence course project, Prof. Shahrouz Moaven, Spring 2014.

REFERENCES

- Professor Rajiv Gupta (My Ph.D. Supervisor)
 - Email: gupta@cs.ucr.edu
 - Homepage: www.cs.ucr.edu/~gupta
 - Professor Nael Abu-Ghazaleh (My Ph.D. Supervisor)
 - Email: naelag@ucr.edu
 - Homepage: www.cs.ucr.edu/~nael
 - Professor Shaahin Hessabi (My M.Sc. Supervisor)
 - Email: hessabi@sharif.edu
 - Homepage: sharif.edu/~hessabi
-