

# Armin Gerami

+1 667 200 9253 | [agerami@umd.edu](mailto:agerami@umd.edu)  
[in/armingerami](https://github.com/armingerami) | [github.com/armingerami](https://github.com/armingerami)

## EDUCATION

---

### University of Maryland

M. Sc. in Telecommunications (Electrical Engineering)

- GPA: 3.8

College Park, MD

January 2022 - May 2023 (*Expected*)

### Sharif University of Technology

B. Sc. in Electrical Engineering

- GPA: 16.33/20

Tehran, Iran

September 2016 - December 2020

## AWARDS AND HONORS

---

### Ranked 21st in Iran's National University Entrance Exam (Konkour) in 2016.

Among 250,000 students

## EXPERIENCE

---

### University of Maryland, Software Developer

January 2022 – Present

#### Creator of The "Rapid Audit" Software

*Description:* Using python, I made a software that calculates various criteria such as energy, money and greenhouse gas reduction potential and visualizes the result as a fully interactive plot. Other outputs are ranking of buildings as an Excel file or summary report in PDF format. The input files are energy bills. (The plot can be accessed [here](#) and the guide can be accessed [here](#)).

### Member of ONRL Research Lab, Sharif University of Technology

July 2021 – December 2021

#### Research on CDMA Quantum Optical Communications.

*Description:* A quantum signal is a mixed quantum state which is a superposition of multiple pure or statistically mixed states. Depending on the operator we use to observe the signal, the output probability distribution will be different. This fact can be used to map each pure state to a user. Using quantum coding theory, we need to find QECC codes with a tradeoff between probability of error and rate. (more info: [Quantum CDMA Communication Systems](#))

### Member of DSP Research Lab, Sharif University of Technology

September 2019 – December 2019

#### Learning a directed Graph through Graph Signal Processing.

*Description:* Following the work in the [Graph Signal Processing: Overview, Challenges, and Applications](#) paper, given a set of Graph Signals and assuming the graph is smooth and undirected, the graph can be learnt through Graph Signal Processing. However, if the graph is directed, we can only learn the graph up to knowing whether or not an edge exists. I have used a Dynamic Programming algorithm to find a combination of edge directions that fits the input signals in  $O(n^3)$ , where  $n$  is number of edges (This was my B.Sc. project). (more info: [Github](#))

### Intern, Iran Telecommunication Research Center

June 2019 – August 2019

#### VHDL Design

*Description:* Writing the Flow-Table Controller (an IP based command look-up table) of an Open-Flow switch in an optical communication network using VHDL. Our hardware was Xilinx xc7a200tfg484 FPGA, our environment was Vivado. (more info: [Github](#))

## OTHER PROJECTS

---

### Sugiyama Decoding Algorithm

*Description:* The Sugiyama decoding algorithm is for [BCH codes](#). In this project, the code is a  $[127, 99, 9]$  ECC (in other words a Finite Field based on  $2^{127}$  with  $2^{99}$  codes which have a minimum distance of 9). since instead of numbers we have power series as elements of the Field, the arithmetic has to be redefined from scratch. The way I achieved this was to use the fact that each element can be uniquely represented by a power of the field generator, and I kept track of the powers. The arithmetic which I defined were addition, multiplication, division and logarithm, and to deploy them efficiently I used theorems such as [Frobenius endomorphism](#)). (more info: [Github](#))

### Simulating Cellular Communications Performance

*Description:* Considering building blockage, hand-off, propagation and diffraction loss, shadowing and fading criteria such as number of successful and failed connections and S/I was calculated in a suburban area of  $25 \text{ km}^2$  (more info: [Github](#))

### Messenger

*Description:* Using Java, I made a script which allows two people in the same network to connect and chat with each other. (more info: [Github](#))

### Analytical Comparison of Various Fiber-Optic CDMA Structures

*Description:* Using Matlab, I simulated performance of "Double optical hard-limiter + passive correlator" and "Double optical hard-limiter + double correlator" channels. (more info: [Github](#))

### Implementing Various Sorting Methods Of Sorting On FPGA

*Description:* Various sorting methods such as Bucket-Sort, Merge-Sort and Bubble-Sort were implemented on a FPGA using Verilog.

### Simple Minesweeper

*Description:* Using C, I created a simple minesweeper. The platform was in the terminal, and to squares could be chosen by inputting text at each step.

## SELECTED COURSES

---

<b>Networks and Protocols</b>   <i>Graduate level</i>	Fall 2022	In Progress
<b>Cellular Communication</b>   <i>Graduate level</i>	Spring 2022	Grade: A
<b>Classical and Quantum Coding Theory</b>   <i>Graduate level</i>	Spring 2022	Grade: A-
<b>Quantum Signals</b>   <i>Graduate level</i>	Fall 2020	Grade: 18.5/20
<b>Algorithms and Data Structures</b>   <i>Undergraduate level</i>	Spring 2020	Grade: 18/20
<b>Causal Inference</b>   <i>Graduate level</i>	Fall 2019	Grade: 17.5/20
<b>Bitcoin and Blockchain</b>   <i>Graduate level</i>	Fall 2019	Grade: 17/20
<b>Machine Learning</b>   <i>Undergraduate level</i>	Fall 2019	Grade: 18.5/20
<b>Optical Communication Networks</b>   <i>Graduate level</i>	Spring 2019	Grade: 17/20

## TECHNICAL SKILLS

---

### Coding Languages:

**Expert:** C++, Matlab, Python, SystemVerilog, Verilog.

**Comfortable:** C, Java, R, VHDL.

**Familiar:** Assembly, CSS, HTML, Javascript, Kotlin, Latex, Remix.

**Software:** Altium Designer, Analytic Solver, Android Studio, AutoCAD, HSPICE, ISE Designer, ModelSim, Microsoft Excel, Sagemath, SIMATIC, Vivado, Wireshark.

**Other:** Linux System.