# DEJAN GRUBISIC

+1 832-938-7867 | dejan.grubisic@rice.edu |
https://dejangrubisic.github.io

## EDUCATION

**Doctor of Philosophy** | ***High Performance Computing and Machine Learning*** Aug. 2019 – May 2024
Rice University | GPA: 3.56 Houston, TX

**Master of Science** | ***Big Data Architectures*** Aug. 2018 – May 2019
University of Novi Sad | GPA: 4.00 Novi Sad, Serbia

**Bachelor of Science** | ***Electrical and Computer Engineering*** Aug. 2014 – May 2018
University of Novi Sad | GPA: 3.96 Novi Sad, Serbia

For all passed courses check here All Courses

## INTERNSHIPS AND WORK EXPERIENCE

**Rice University** Aug. 2019 – May 2024
Graduate student Houston, TX

- Developing GPU support for large-scale profiler - HPCToolkit
- Creating analysis techniques for GPU accelerated applications
- Developing profiler guided optimizations based on reinforcement learning - Compiler2

**Meta AI** May 2023 – Dec. 2023
Machine Learning Research Intern Menlo Park, CA

- Finetuning Llama2-based model on LLVM IR programs to solve phase-ordering problem
- Analyzing LLM's capability in applying 60 LLVM optimizations

**Meta AI** May 2022 – Dec. 2022
Machine Learning Research Intern New York City, NY

- Building reinforcement learning framework - LoopTune for optimizing tensor operations
- Building and tuning cost and policy deep learning models

**Berkeley Lab** Jun. 2021 – Sep. 2021
High Performance Computing Research Intern Berkeley, California

- Profiling and analysis of power consumption on multi node GPU applications

**Rice University** Jun. 2018 – Sep. 2018
Research intern Houston, TX

- Optimization of MADNESS (Multiresolution Adaptive Numerical Environment for Scientific Simulation) in Intel CnC workframe

**Institute for High Performance Microelectronics** Jun. 2017 – Sep. 2017
Hardware Engineer Intern Frankfurt O, Germany

- Profiling and Analysis of FFT implementation on Xtensa Platform in C
- Dhrystone Benchmark for FFT and theoretical analysis window functions

## RESEARCH AND PROJECTS

**Statistical Machine Learning**
Kaggle challenge: NBME - Score Clinical Patient Notes

- Fine tuning Bert model to predict features from the patient notes ( Python )

**Artificial Inteligence**
Pacwar: Finding the strongest gene by using genetic algorithms

- Hill climbing, genetic algorithm, finding local hills with K-means, defining scoring function and other tricks in ( Python )

**Multiprocessing**
Lock free concurrent skip list
- Implemented using compare-and-swap atomic primitives and OpenMP ( C )

**Parallel Computing**
Parallel algorithms in various technologies
- Parallel exploratory search of game Othelo using Cilk ( C++ )
- LU decomposition using OpenMP ( C++ )
- 2.5 matrix multiplication using OpenMPI ( C++ )
- Bitonic sort using Cuda (CudaC++)

**Compiler Construction 2**
Compiler for DEMOgram language
- Compiler for custom language implemented in Flex/Bison framework ( C )

**Compiler Construction 1**
Compiler for ILOC language
- Implementating scanner, parser, registar allocator and instruction schedulers implemented ( C++ )

**Master Thesis**
Finding multi-source shortest path in dynamic large-scale graph, based on Lambda architecture
- Implementing batch and real-time processing algorithms ( pySpark )
- Implementing storage in HDFS and communication in Kafka framework
- Creating web based application ( Pyton Dash )
- Using Docker for containerization

**Data Science**
Movie profit prediction
- Predicting audience interest in movie by using logistic regression, k-nearest neighbor, support vector machines
- Predicting weekend profit for a movie by using linear regression, neural network, SVM and NaiveBayes

**Bachelor Thesis**
FPGA design of hardware core for acceleration of chess engine
- Analysis and design of software-hardware system using ESL methodology ( SystemC )
- Implementing board evaluation module in VHDL
- Verification of design using UML methodology (SystemVerilog)

**Applied Electronics**
Metal Detector
- End-to-end implementation of circut, PCB and packing with AltiumDesigner

## PUBLICATIONS

**Large Language Models for Compiler Optimization** Sep. 2023
Preprint available on Arxiv

**LoopTune: Optimizing Tensor Programs with Reinforcement Learning** Sep. 2023
Preprint available on Arxiv

**LoopStack: ML-friendly ML Compiler Stack** Nov. 2022
Neural Information Processing Systems

**Measurement and Analysis of GPU-Accelerated OpenCL Computations on Intel GPUs** Nov. 2021
International Workshop on Programming and Performance Visualization Tools

**An Automated Tool for Analysis and Tuning of GPU-accelerated Code in HPC Applications** Feb. 2021
IEEE Transactions on Parallel and Distributed Systems

**Measurement and Analysis of GPU-accelerated Applications with HpcToolkit** Nov. 2020
Parallel Computing Journal

**Finding multi-source shortest path in dynamic large-scale graph in Lambda architecture** May 2019
Faculty of Technical Sciences in Novi Sad Journal

## SKILLS

**Programming Languages**: C/C++, Python, CudaC, Java, Bash, VHDL

**Technologies**: Pthreads, OpenMP, MPI, Docker, Slurm, Spark, Linux, PyTorch

**Featured Skills**: Parallel Computing, Compiler Construction, Profiling Tools