

Francisco Pereira

✉ francisco.chamica.pereira@tecnico.ulisboa.pt 🌐 github.com/fchamicapereira 🌐 https://fchamicapereira.me

Office 601, INESC-ID, Rua Alves Redol 9, 1000-029, Lisbon, Portugal

Last updated: August 1, 2023

Education

Instituto Superior Técnico (University of Lisbon) 2021 - Present

Ph.D. in Computer Science and Engineering

Advisors: Prof. Luis Pedrosa, Prof. Fernando Ramos, Prof. Justine Sherry

Instituto Superior Técnico (University of Lisbon) 2019 - 2021

M.Sc. in Computer Science and Engineering

Advisors: Prof. Luis Pedrosa, Prof. Fernando Ramos

Thesis: Parallel Network Function Synthesis

Instituto Superior Técnico (University of Lisbon) 2016 - 2019

B.Sc. in Computer Science and Engineering

Research Interests

Networking and Computer Systems. Automating performance optimizations to Network Functions. Building compilers for automatic offloading of Network Functions to hardware network accelerators (FPGAs, SmartNICs, and Programmable Switches such as the Intel Tofino).

Scholarships and Awards

- CMU Portugal Affiliate Ph.D. Program, FCT (Fundação para as Ciências e Tecnologias) 2021 - Present
- Master Student Merit Award 2020, Distributed Systems Group @ INESC-ID 2020

Publications

Automatic Parallelization of Software Network Functions 2024

Francisco Pereira, Fernando Ramos, Luis Pedrosa.

21st USENIX Symposium on Networked Systems Design and Implementation (**NSDI'24**).

Automatic Generation of Network Function Accelerators Using Component-Based Synthesis 2022

Francisco Pereira, Gonçalo Matos, Hugo Sadok, Daehyeok Kim, Ruben Martins, Justine Sherry, Fernando Ramos, Luis Pedrosa.

ACM SIGCOMM Symposium on SDN Research (**SOSR'22**).

Selected Posters

In-Network ML Feature Computation for Malicious Traffic Detection 2023

João Amado, **Francisco Pereira**, Salvatore Signorello, Miguel Correia, Fernando Ramos.

ACM SIGCOMM'23 Posters and Demos.

Projects

SyNAPSE, DPSS @ INESC-ID & SNAP Lab @ CMU 2021 - Present

When implementing network functions (NFs), developers are often confronted with a choice: implement the NF in software and face the challenge of performance, or use one of a wide variety of programmable networking devices, such as programmable switches and SmartNICs, to trade-off some flexibility for the ability to process

packets at full line rate. With SyNAPSE we ask “Why not have both?”. We propose a synthesis based approach to automatically generate accelerated implementations of a Software NFs, using smart network devices whenever possible to increase performance.

Maestro, DPSS @ INESC-ID

2020 - 2023

Software network functions (NFs) trade-off flexibility and ease of deployment for an increased challenge of performance. Fortunately, a number of tools, techniques, and architectural enhancements have come together over the years to enable packet processing at 10s Gbps, but achieving these traffic rates requires a deep understanding of low-level details. The traditional way to increase NF performance is by distributing traffic to multiple CPU cores, but this poses a significant challenge: *how to parallelize an NF without breaking its semantics?* We propose Maestro, a tool that analyzes a sequential implementation of an NF and automatically generates an enhanced parallel version.

Teaching Experience

Distributed Systems

Spring 2022

Instituto Superior Técnico (University of Lisbon)
Undergraduate Teaching Assistance

Computer Networks

Fall 2022

Instituto Superior Técnico (University of Lisbon)
Undergraduate Teaching Assistance

Computer Networks

Fall 2021

Instituto Superior Técnico (University of Lisbon)
Undergraduate Teaching Assistance

Community Service

Organization Committee, Netdev 0x16

2022

HotCRP Chair, EuroP4'21

2021

Network admin, DPSS @ INESC-ID

2021 - Present

Student Volunteer, SIGCOMM'21

2021

DevTeam Team Leader, SINFO

2018 - 2020

Programming Skills

Programming languages

Daily use: C, C++, Python, P4

Comfortable: Javascript, Java

Used in the past: C#, Go

Frameworks

DPDK, KLEE, LLVM, Z3, node.js, AngularJS

Other tools and libraries

Docker, vagrant, gdb, git, bash, gnuplot, matplotlib, libpcap, scapy