Curriculum Vitae

André Martins Pereira

August 1, 2022

Notice

The information presented in this Curriculum Vitae has been organised and distributed according to the criteria presented in **Public Notice 869/2022**, for a position of Assistant Professor in the subject area of Informatics at the University of Minho.

All blue highlighted text in the digital version of this document links to specific document sections or external hyperlinks. Each section is accompanied by the relevant tags from the ones listed under the section **10.1** of the **Public Notice 869/2022**.

Contents

Assinado por: **ANDRÉ MARTINS PEREIRA**
Num. de Identificação: 13773075
Data: 2022.08.01 13:39:54 +0100

(André Martins Pereira)

André Martins Pereira, **PhD**

Post-Doc Researcher | Invited Assistant Professor

linkedin.com/in/ampereira90 +351 934 855 058
https://ampereira90.github.io ampereira90@gmail.com ampereira@di.uminho.pt
Rua do Mercado Municipal N35-314, 4720-086 Amares, Portugal
Sex Male | **Date of Birth** 19/12/1990 | **Nationality** Portuguese | **Languages** English/Portuguese

[ORCID 0000-0002-2110-914X] [Researcher ID AAU-4180-2021] [Google Scholar] [Scopus ID 57221290901] [DBLP]

Education

2013 - 2019 **Doctoral Degree in Informatics Engineering** (MAP-i) Final grade of *Very Good*

Dissertation subject: "*HEP-Frame*: a Development Aid and Efficient Execution Engine where a Multi-layer Scheduler Adaptively Orders Pipelined Data Stream Applications", supervised by Prof. Alberto Proença and Prof. António Onofre.

Research interests focused on the development of a framework for portable and efficient coding and execution of pipelined data analysis applications on heterogeneous servers with multicore, manycore, and accelerator devices, which incorporates software engineering principles on the development workflows of non-computer scientists. Additional work on the efficient execution of neural networks (both shallow and deep) on homogeneous servers and embedded systems.

Courses attended during the doctoral programme:

- Adaptive Business Intelligence.
- Cryptography and Information Security.
- Knowledge Discovery from Databases.
- Source Code Analysis and Manipulation.

The work was partially funded by *FCT*, through the grant SFRH/BD/119398/2016, among other projects.

2011 - 2013 **Masters in Informatics Engineering** (University of Minho) Final grade of 17 out of 20.

Dissertation subject: "Efficient Processing of ATLAS Events Analysis on Homogeneous and Heterogeneous Platforms with Accelerator Devices", supervised by Prof. Alberto Proença and Prof. António Onofre.

Specialised in Parallel and Distributed Computing and Computer Graphics:

- Computer Systems and Performance.
- Parallel Computing Paradigms.
- Numerical Methods and Algorithms.
- Computer Vision.
- Modelling and Visualisation.
- Illumination and Photo-realism.
- Virtual Augmented Reality.

2008 - 2011 **Bachelor in Informatics Engineering** (University of Minho) Final grade of 13 out of 20.

- Programming Paradigms (functional, imperative, object-oriented).
- Math (discrete, calculus, logic, linear algebra and numerical methods).
- Systems Engineering (economy and cost analysis, statistics, operational research).
- Computer Architecture, Operating and Distributed Systems, and Networks.
- Software Engineering (algorithms and complexity, software systems architecture and design).
- Physics (modern topics, electromagnetism).
- Databases and Knowledge Representation and Reasoning.

Advanced Training

2022 Minho Advanced Computing Center User Group Workshop, at the Crowne Plaza, in Porto, Portugal, from July 7^{th} to 8^{th}.

2021 Minho Advanced Computing Center User Group Workshop, *online only*, from June 15^{th} to 18^{th}.

2017 Workshop on Advanced Computing for Earth Sciences, at the University of Porto, Portugal, from December 18^{th} to 20^{th}.

2016 TACC Summer School in Advanced Scientific Computing, in Braga, Portugal, from June 20^{th} to 23^{rd}.

2014 37^{th} CERN School of Computing, in Braga, Portugal, from August 25^{th} to September 6^{th}.

2014 Summer School on Parallel High Performance Computing using Accelerators, in Braga, Portugal, from June 25^{th} to 27^{th}.

2014 MAP-i Spring School on Logic of Dynamical Systems, in Braga, Portugal, from March 24^{th} to 28^{th}.

2014 4^{th} International Doctorate Network in Particle Physics, Astrophysics and Cosmology (IDPASC) Winter School, in Braga, Portugal, from January 20^{th} to 28^{th}.

Professional Experience

Present
August 2022

Post-Doc Researcher, HASLAB AT THE UNIVERSITY OF MINHO, Portugal
High Performance Computing (HPC) Specialist for THE MINHO ADVANCED COMPUTING CENTER (*MACC*)
Responsible for:

- Improving the scalability and performance of *HPC* scientific tools and applications of *MACC* industry partners.
- Configuring, managing, and utilising tools for the profiling, debugging, and visualisation of performance reports of parallel applications.
- Developing, assessing, and maintaining mechanisms for the collaboration with industry partners, focusing on coding best practices, transfer of know-how, and technological consulting.

High Performance Computing | Scientific Computing | Performance Engineering

Present
October 2019

Invited Assistant Professor, UNIVERSITY OF MINHO, Portugal- CP1
Assistant professor for several curricular units in the field of computing systems and parallel computing. Further details in subsection 5.1.

July 2022 – April 2019

Software Engineer, Electron Softview, Portugal

Lead Software Developer for the TopoSEM project

Agile development of a solution for 3D reconstruction and analysis of Scanning Electron Microscope (*SEM*) images, through the integration of an existing command-line workflow into a final software product. Responsible for:

- Requirements analysis with stakeholders and potential end-users of the software product.
- Design of the TopoSEM core software architecture, considering both online (*SaaS*) and offline (standalone application) usage scenarios.
- Integration of existing *SEM* imaging manipulation algorithms into a polished software package.
- Design and development of proprietary file formats to hold topography data based on the 3D reconstruction of *SEM* images.
- Design and implementation of User Interfaces based on identified workflows.
- Design of the TopoSEM web service (*SaaS*) architecture.
- Management and supervision of the dissertation work by a MSc student integrated in the TopoSEM project software stack.

Software Engineering | Test Driven Development | Qt | C++ | Visual Studio | Docker | Kubernets | SaaS | Agile

September 2019 – March 2014

Invited Teaching Assistant, University of Minho, Portugal- CP1

Teaching assistant for several curricular units in the field of computing systems and parallel computing. Further details in subsection 5.1.

April 2019 – January 2019

Post-Doc Researcher, University of Minho, Portugal

Full-stack Software Engineer for the Reaction DataBase with Transition State (*RDB-TS*) Project

Centred on extending a web platform prototype, which encompasses the transition state database, molecule rendering, and SMILE-based search modules.

- Architecture design and implementation of database models to hold reaction transition states data.
- Extension of the online molecule 3D rendering web service.

Software Engineering | Docker | Kubernets | SaaS | PostgreSQL | Node.js | WebGL

December 2018 – February 2014

Researcher, University of Minho, Portugal

Parallel Computing Specialist for LIP-Minho Associated Laboratory and Algoritmi Research Centre

Design and deployment of the *HEP-Frame* software tool, a framework to aid scientists to develop applications to analyse data from a large datasets, with a flexible pipeline of tasks. This framework, targeted at non-computer scientists, aims to provide efficient parallel code execution without requiring user expertise in parallel computing. It aids the development scientific code, while subtly imposing restrictions for users to code according to software engineering principles (in algorithm design and code structuring).

- Design and development of workload schedulers based on novel load balancing strategies for inter- and intra-server parallel applications.
- Design and development of custom parallel solutions for *GPU* and manycore hardware accelerators.
- Design and development of a feature-rich framework, currently in use by multiple research groups.
- Identification of design problems in existing pipelined scientific code and analysis of the relevant software engineering principles key to produce computationally efficient, maintainable, and long lasting code bases.
- Close integration in a multi-disciplinary team of computer scientists and particle physicists.
- Administration of Linux-based research servers: user management and software deployment.

High Performance Computing | Scientific Computing | Data Analysis | Stream Computing

March 2016 – October 2015 | **Invited Teaching Assistant, Instituto Politécnico do Cávado e Ave, Portugal**- OAR1
Responsible for the restructuring of the *Arquitetura de Dispositivos Móveis* curricular unit of the *Desenvolvimento de Aplicações Móveis* professional degree. Further details in subsection 7.1.

January 2014 – January 2013 | **Junior Researcher, University of Minho, Portugal**
Parallel Computing Engineer for LIP-Minho
Performance analysis and improvement of a particle collision data analysis application.
- Profiling, analysis, and characterisation of the software bottlenecks.
- Development of custom solutions for the performance improvement of code sections in homogeneous (shared memory) and heterogeneous (distributed memory) servers.

High Performance Computing | Scientific Computing | CUDA | Accelerator Devices | OpenMP

August 2012 – July 2012 | **Research Intern, University of Texas at Austin, USA** - DC4
Parallel Computing Engineer for Institute of Computational Engineering and Sciences
Responsible for the optimisation and parallelisation of a molecular docking scientific code on heterogeneous servers with *GPUs*. Further details in subsection 4.4.
High Performance Computing | Scientific Computing | CUDA | Accelerator Devices | MPI

Scientific Contributions

4.1 Scientific Performance - DC1

The list of accepted publications can be found below. A brief summary of their venue classifications, based on the ScimagoJR/Core ranking systems, is also included. Note that I was the main author in all publications. The citation count was obtained by cross referencing Google Scholar and Scopus data, and includes self-citations. These publications, along with other metrics, can also be found in Google Scholar.

The publications considered most representative of my research activity are marked with ★. A brief justification of their scientific relevance is provided in a separate document in the document stack that this CV is part of.

Publications

1 ★ 2021 André Pereira and Alberto Proença
HEP-Frame: Improving the Efficiency of Pipelined Data Transformation & Filtering for Scientific Analyses
Computer Physics Communications, Volume 263, Elsevier.
Q1 Journal with an impact score of 6.55 and h-195 index in Scimago.
Indexed by WoS and Scopus.
Access the paper.
Citations: 1

2 2020 André Pereira and Alberto Proença
PRNG-Broker: A High-Performance Broker to Supply Parallel Streams of Pseudo-Random Numbers for Large Scale Simulations
Technical report. Currently under submission to a conference.
Access the report.

3 ★ 2018 André Pereira and Alberto Proença
Efficient Use of Parallel PRNGs on Heterogeneous Servers
In Proceedings of the International Conference on Mathematical Applications, pp. 7–12, Institute of Knowledge and Development.
Indexed by WoS.
Access the paper.

4 ★ 2016 André Pereira, António Onofre, and Alberto Proença
Tuning Pipelined Scientific Data Analyses for Efficient Multicore Execution
In Proceedings of the International Conference on High Performance Computing & Simulation, pp. 751-758, IEEE. *Invited for a submission of an extended version of the publication.*
Impact score of 1.98 (as of 2020) and h-11 index in Scimago.
Indexed by WoS and Scopus.
Access the paper.
Citations: 4

5 ★ 2015 André Pereira, António Onofre, and Alberto Proença
HEP-Frame: A Software Engineered Framework to Aid the Development and Efficient Execution of Scientific Code
In Proceedings of the International Conference on Computational Science and Computational Intelligence, IEEE.
Impact factor of 0.161 and h-9 index in Scimago.
Indexed by WoS and Scopus.
Access the paper.
Citations: 2

6 ★ 2014 André Pereira, António Onofre, and Alberto Proença
Removing Inefficiencies from Scientific Code: The Study of the Higgs Boson Couplings to Top Quarks
In Proceedings of the International Conference on Computational Science and its Applications, LNCS vol.8582, Part IV, pp. 576–591, Springer.
Impact score of 1.16 (as of 2021) and h-6 index in Scimago.
Indexed by WoS and Scopus.
Access the paper.
Citations: 2

Participation in International Scientific Events

Minho Advanced Computing Center User Group Workshop (*MUG* 2022)
July 7^{th} *to* 8^{th}*, 2022, Porto, Portugal*

Minho Advanced Computing Center User Group Workshop (*MUG* 2021)
June 15^{th} *to* 18^{th}*, 2021, Portugal* (online event)

The 2020 World Congress in Computer Science, Computer Engineering, & Applied Computing (*CSCE*)
July 27^{th} *to* 30^{th}*, 2020, Las Vegas, NV, USA* (virtual participation)

The 2017 Workshop on Advanced Computing for Earth Sciences (*ACES*)
December 18^{th} *to* 20^{th}*, 2017, Porto, Portugal*

The 2016 International Conference on High Performance Computing & Simulation (*HPCS*)
July 18^{th} *to* 22^{nd}*, 2014, Innsbruck, Austria*

TACC Summer School in Advanced Scientific Computing
June 20^{th} *to* 23^{rd}*, 2016, Braga, Portugal*

The 2015 International Conference on Computational Science and Computational Intelligence (*CSCI*)
December 7^{th} *to* 9^{th}*, 2015, Las Vegas, NV, USA*

The Inverted CERN School of Computing (*iCSC*)
February 23^{rd} *to* 24^{th}*, 2015, CERN, Switzerland*

The 37^{th} CERN School of Computing (*CSC*)
August 24^{th} *to September* 6^{th}*, 2014, Braga, Portugal*

The 14^{th} International Conference on Computational Science and Its Applications (*ICCSA*)
June 30^{th} *to July* 3^{rd}*, 2014, Guimarães, Portugal*

Summer School on Parallel High Performance Computing using Accelerators
June 25^{th} *to* 27^{th}*, 2014, Braga, Portugal*

The 4^{th} International Doctorate Network in Particle Physics, Astrophysics and Cosmology (*IDPASC*)
January 20^{th} *to* 28^{th} *2014, Braga, Portugal*

Tools

HEP-Frame is a framework to aid the development and efficient execution of pipelined data analysis applications in homogeneous and heterogeneous servers. This tool resulted from the work developed during my *PhD*, and is currently actively used in particle physics research. The scientific details of *HEP-Frame's* core components are described in publication #**1**. The tool is publicly available at https://bitbucket.org/ampereira/hep-frame/wiki/Home.

PRNG-Broker is a middle layer between the application code, e.g., a Monte Carlo simulation, and specialised pseudo-random number generation (*PRNG*) libraries. It efficiently manages parallel *PRN* requests to external *PRNG* libraries, adequately using the computational resources available in multicore, manycore, and *GPU* devices. This efficient management of *PRN* generation focus on improving the performance of parallel compute-bound applications, but also provides a significant benefit for both sequential and memory-bound codes. The technical report #**2** provides a description of the *PRNG-Broker*. The tool is publicly available at https://github.com/prng-broker/prng-broker/wiki/PRNG-Broker.

4.2 Special Lectures - DC2

February 2015 Invited lecturer at the Inverted CERN School of Computing (*iCSC*) at CERN. Responsible for the planing, creation of the material, and lecturing a 4-hour session on the development of efficient parallel code, whose target audience academic background was in either experimental or theoretical physics. These scientists, while extremely specialised in their fields of study, were mostly self-taught programmers with little to no expertise in Computer Science.

4.3 Involvement in Scientific Projects and Grants - DC2, DC3, DC4

Phenomenological Studies at the LHC

Co-funded by FCT and the CERN Fund
Reference: CERN/FP/123619/2011
Junior Researcher from 2013 to 2014

Plano Estratégico da Escola de Engenharia - Agenda 2020: Projetos multidisciplinares

Funded by the Schoolf of Engineering of the University of Minho (EEUM)
Doctoral Grant holder from 2014 to 2015

Laboratório de Instrumentação e Física Experimental de Partículas (LIP-Minho)

Funded by FCT/MEC
Reference: UID/FIS/50007/2013
Researcher from 2015 to 2016

FCT PhD Grant

Funded by FCT
Reference: SFRH/BD/119398/2016
Doctoral Grant holder from 2016 to 2019

RDB-TS: A Reaction DataBase with Transition State information from quantum chemical calculations

Funded by FCT
Reference: NORTE-01-0145-FEDER-031689
Post-doctoral Researcher in 2019

EUROCC: National Competence Centres in the framework of EuroHPC

Funded by FCT
Reference: 951732-H2020-JTIEuroHPC-2019-2
Post-doctoral Researcher since 2022

4.4 Internship in International Institutes - DC4

August 2012 – July 2012

Research Intern, University of Texas at Austin, USA
Parallel Computing Engineer for Institute of Computational Engineering and Sciences
Responsible for the parallelisation of *F2Dock*, a software package for protein-protein docking, on distributed memory environment servers with *GPU* devices. Integrated in a research group under the supervision of professor Chandrajit Bajaj. Work focused on:

- Profiling, analysis, and characterisation of the software bottlenecks.
- Design and implementation of parallelisation strategies to speedup the execution time of specific code sections for multi-*GPU* servers.

5.1 Lecturing Activities - CP1

Present – October 2019 | **Invited Assistant Professor, University of Minho, Portugal**

- *Sistemas da Computação*, for *Licenciatura em Engenharia Informática*, *Mestrado Integrado em Engenharia Informática*, *Mestrado Integrado em Engenharia Física*, and *Licenciatura em Ciências da Computação*. Academic years: 19/20, 20/21, and 21/22.
- *Arquitetura de Computadores*, for *Licenciatura em Engenharia Informática* and *Mestrado Integrado em Engenharia Informática*. A.y.: 20/21.
- *Arquiteturas Avançadas*, for *Mestrado em Engenharia Informática* and *Mestrado Integrado em Engenharia Informática*. A.y.: 20/21.
- *Programação Paralela*, for *Mestrado Integrado em Engenharia Física* and *Licenciatura em Engenharia Física*. A.y.: 21/22 and 22/23.
- *Computação Paralela*, for *Mestrado Integrado em Engenharia Informática* and *Mestrado em Engenharia Informática*. A.y.: 21/22 and 22/23.

September 2019 – March 2014 | **Invited Teaching Assistant, University of Minho, Portugal**

- *Sistemas da Computação*, for *Licenciatura em Engenharia Informática*, *Mestrado Integrado em Engenharia Informática*, and *Mestrado Integrado em Engenharia Física*. Academic years: 14/15, 15/16, 16/17, 17/18, and 18/19.
- *Arquitetura Avançada de Computadores*, for *Mestrado em Engenharia Informática*. A.y.: 14/15.
- *Arquitetura de Computadores*, for *Licenciatura em Engenharia Informática* and *Mestrado Integrado em Engenharia Informática*. A.y.: 18/19 and 19/20.
- *Arquiteturas Avançadas*, for *Mestrado em Engenharia Informática* and *Mestrado Integrado em Engenharia Informática*. A.y.: 16/17.
- *Paradigmas de Computação Paralela*, for *Mestrado em Engenharia Informática*. A.y.: 16/17.

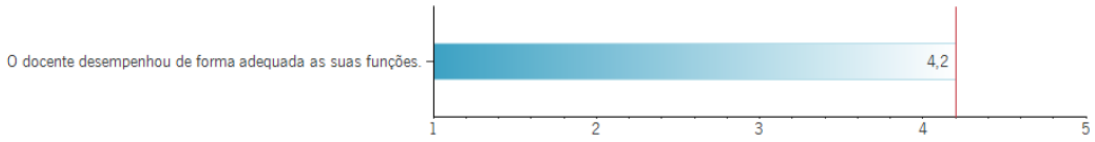
5.2 Pedagogic Performance - CP2

This section includes a summary of my pedagogic performance while lecturing at the University of Minho. The full reports for each Curricular Unit, including students' qualitative feedback, are available at https://ampereira90.github.io/terms/.

ME78ME7800000097_1 - Computação Paralela (1^{st} year)

Year: 2021/2022

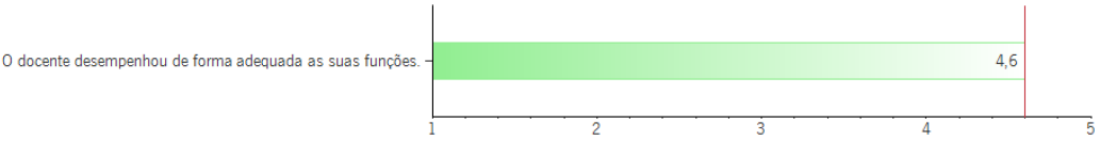
Course: Mestrado em Engenharia Informática



J905N5_1 - Programação Paralela (3^{rd} year)

Year: 2021/2022

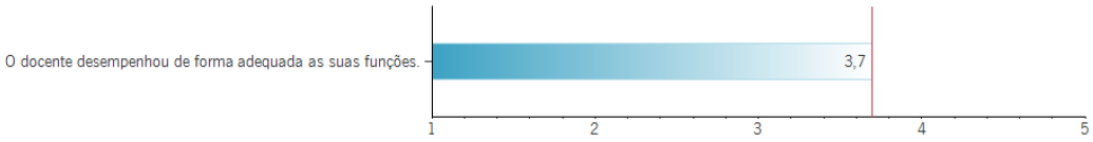
Course: Licenciatura em Engenharia Física



H507U6_1 - Computação Paralela (4^{th} year)

Year: 2021/2022

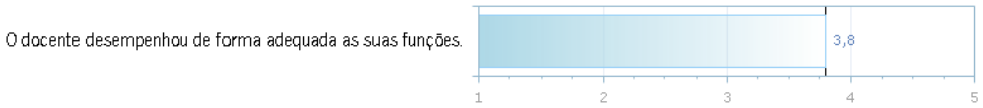
Course: Mestrado Integrado em Engenharia Informática



H502N1_2 - Sistemas de Computação (1^{st} year)

Year: 2020/2021

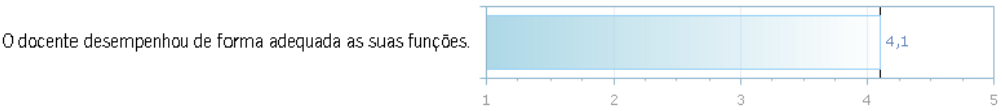
Course: Mestrado Integrado em Engenharia Informática



G504N4_2 - Sistemas de Computação (2^{nd} year)

Year: 2020/2021

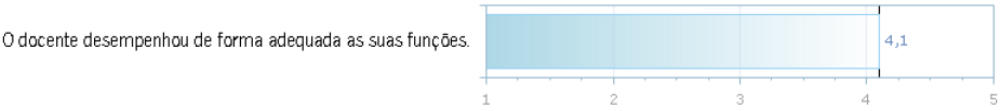
Course: Mestrado Integrado em Engenharia Física



H503N4_1 - Arquitetura de Computadores (2^{nd} year)

Year: 2020/2021

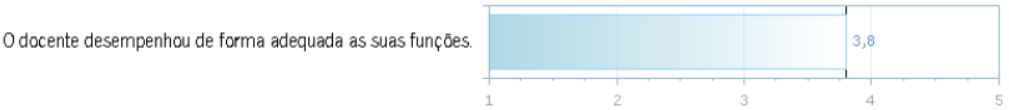
Course: Mestrado Integrado em Engenharia Informática



H502N1_2 - Sistemas de Computação (1^{st} year)

Year: 2019/2020

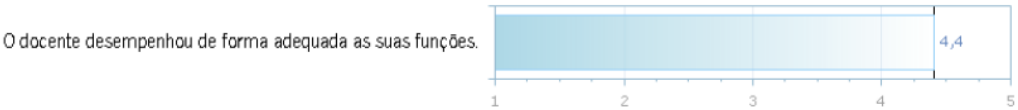
Course: Mestrado Integrado em Engenharia Informática



H503N4_1 - Arquitetura de Computadores (2^{nd} year)

Year: 2019/2020

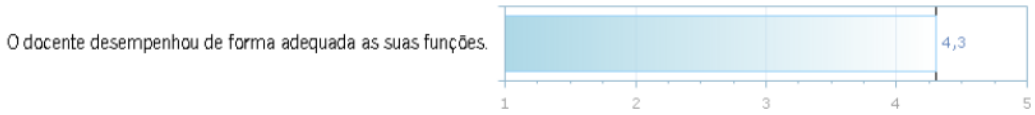
Course: Mestrado Integrado em Engenharia Informática



H502N1_2 - Sistemas de Computação (1^{st} year)

Year: 2018/2019

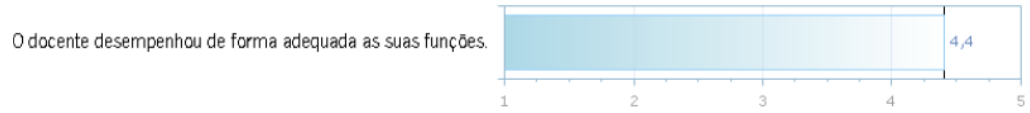
Course: Mestrado Integrado em Engenharia Informática



H503N4_1 - Arquitetura de Computadores (2^{nd} year)

Year: 2018/2019

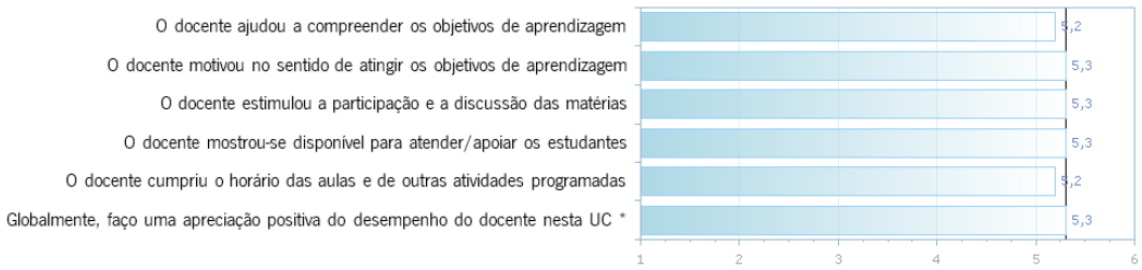
Course: Mestrado Integrado em Engenharia Informática



H502N1_2 - Sistemas de Computação (1^{st} year)

Year: 2017/2018

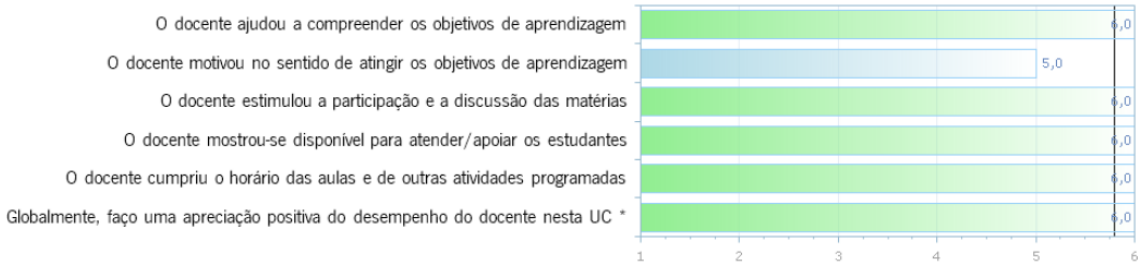
Course: Mestrado Integrado em Engenharia Informática



H507R3_1 - Computação Paralela (4^{th} year)

Year: 2017/2018

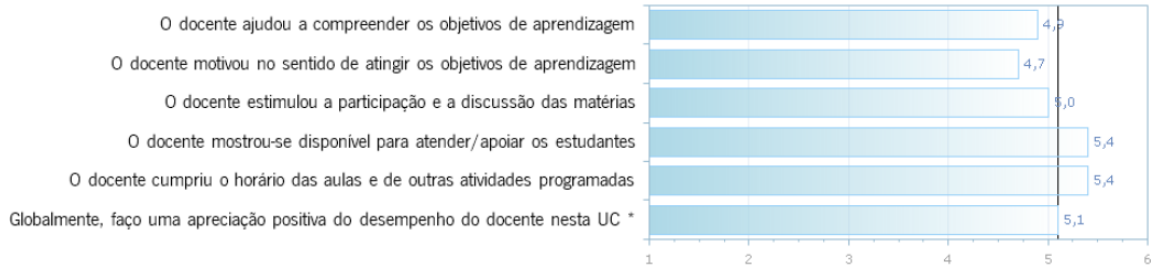
Course: Mestrado Integrado em Engenharia Informática



H507N8_1 - Arquiteturas Avançadas (4^{th} year)

Year: 2016/2017

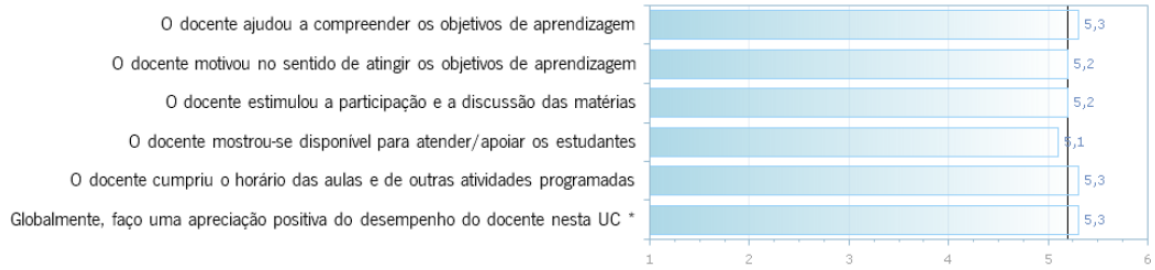
Course: Mestrado Integrado em Engenharia Informática



H502N1_2 - Sistemas de Computação (1^{st} year)

Year: 2016/2017

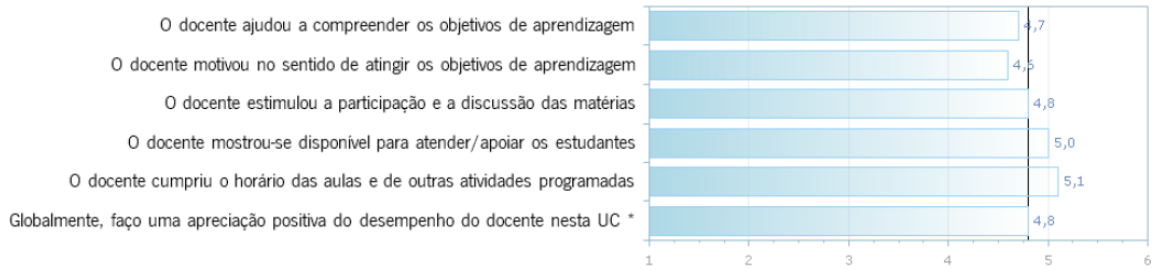
Course: Mestrado Integrado em Engenharia Informática



H507P5_1 - Paradigmas de Computação Paralela (4^{th} year)

Year: 2016/2017

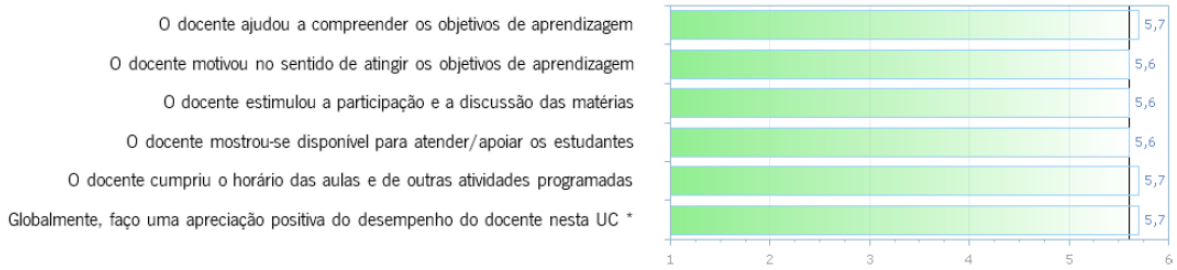
Course: Mestrado Integrado em Engenharia Informática



H502N1_2 - Sistemas de Computação (1^{st} year)

Year: 2015/2016

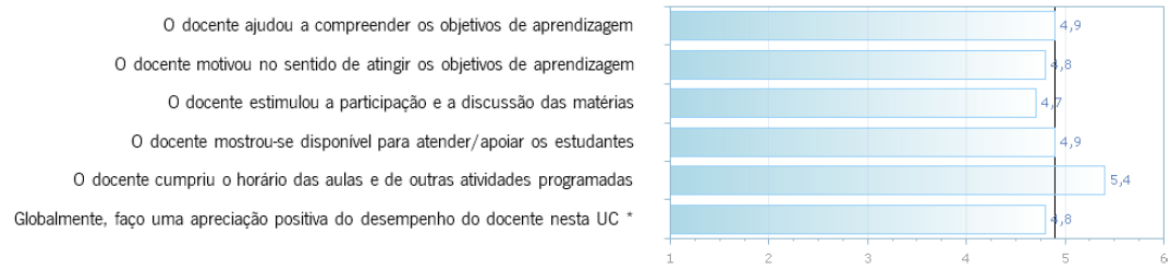
Course: Mestrado Integrado em Engenharia Informática



8202N1_2 - Sistemas de Computação (1^{st} year)

Year: 2014/2015

Course: Licenciatura em Engenharia Informática



5.3 Participation in Pedagogic Projects - CP4, CP5

Responsible for:

- Planing and producing the required content for the lab classes of *Arquiteturas Avançadas*, for the *Mestrado Integrado em Engenharia Informática* and *Mestrado em Engenharia Informática*, along with Professor Alberto Proença (University of Minho). The material ranged from theory-focused presentations of parallel computing paradigms and tools to the guides to be followed by the students during the lab classes.
- Producing content for the lab classes of the new *Computação Paralela* and *Programação Paralela* curricular units, of *Mestrado em Engenharia Informática* and *Licenciatura em Engenharia Física* respectively, lectured since the first semester of 2021/2022. This work is shared with Professor João Sobral and Rui Silva (University of Minho).
- Reviewing and extending the existing content for the lab classes of *Sistemas de Computação*, for the *Mestrado Integrado em Engenharia Informática*.
- Producing content for the lab classes of *Arquitetura de Computadores*, for the *Mestrado Integrado em Engenharia Informática*, along with Professor António Pina (now retired, University of Minho).
- Restructuring and producing content for theoretical and lab classes of *Arquitetura de Dispositivos Móveis*, in 2015/2016, of the *Desenvolvimento de Aplicações Móveis* professional degree, at *Instituto Politécnico do Cávado e Ave*. Further details in subsection 7.1.

Most of the material I produced is available at https://ampereira90.github.io/teaching/.

5.4 Tutoring and Master Thesis Co-Supervisions - CP6

Introduction and training of computer science MSc students on programming accelerator devices - General Purpose *GPUs* and Intel manycore Knights Corner and Knights Landing architectures - applied to scientific fields related to *HPC*, such as particle physics and artificial intelligence. Responsible for tutoring several particle physics BSc and MSc students in the challenges of parallel computing and in using the *HEP-Frame* tool for the development of data analysis code.

Co-supervised *Projetos Integrados*, an academic project for 4^{th} year students in *Mestrado em Engenharia Informática* and *Mestrado Integrado em Engenharia Informática*, which had a duration of 1 to 2 semesters. The students were integrated into existing parallel computing projects, often multi-disciplinary and used real case studies. This work often continued into their MSc dissertations.

2018/2019 "Neural Net Computing in Embedded Systems", work developed with BOSCH on delivering real-time performance of inference process of deep neural networks, by Manuel Moreno and Nuno Magalhães.

2015/2016 "Porting Heterogeneous Schedulers into *HEP-Frame*", focused on the study and integration of *CPU-GPU* load balancing schedulers on the *HEP-Frame* tool, by Nelson Torres and Tiago Santos.

2014/2015 "Performance Evaluation of an ATLAS Experiment Data Analysis Application", work that integrated *HPC* with a real particle physics case study, by John Maia and José Silva.

Co-advisor of several master dissertations in conjunction with different research groups (various multi-disciplinary themes from computer science to physics and chemistry) and BOSCH (focused on real-time issues of *AI* and computer vision computation).

2022 "Improving the Efficiency of Digital Image Correlation Computation for *SEM* Images", to be defended by José Ferreira at the University of Minho, in partnership with Electron Softview.

2022 "Integration of Local and Remote Hardware Accelerators into Parallel and Distributed Applications Using *HPX*", to be defended by Pedro Barbosa at the University of Minho.

2021 "Upgrading the *HEP-Frame* Scheduling Dependency Graph to Support Conditional Task Graphs", work on-hold by José Resende at the University of Minho.

2021 "Optimisation of Deep Learning Algorithms for an Autonomous *RC* Vehicle", work in cooperation with BOSCH, successful defence by André Filipe Pereira at the University of Minho.

2021 "Improving the Efficiency of the Energy-Split Tool to Compute the Energy of Very Large Molecular Systems", successful defence by Sara Pereira at the University of Minho.

2018 "Deeploy: a Neural Network Computer Vision Tool for the NVidia Tegra TX2 Embedded System", work in cooperation with BOSCH, successful defence by João Fernandes at the University of Minho.

2016 "Integrating Heterogeneous Computing Features Into *HEP-Frame*", successful defence by John Maia at the University of Minho.

Scientific and Pedagogic Project

This section presents the scientific-pedagogic project required under section **4.2.c** of **Public Notice 869/2022**, for the recruitment of an Assistant Professor in Informatics for the Department of Informatics of the University of Minho.

Given my scientific and academic background, I believe that I can contribute the most to the department research activities in the field of Parallel Computing of the Public Notice. However, my experience as a post-doc researcher at the *RDB-TS* project and as the lead software engineer of the TopoSEM project at Electron SoftView for more than 2 years developed my expertise to also contribute significantly to the scientific and pedagogic activities within the Software Engineering field.

Subsection 6.1 contextualises both research and pedagogical activities that I aim to pursue. Subsection 6.2 presents the research activities that compose the scientific part of the project. Finally, subsection 6.3 presents the pedagogic activities to be developed in both Software Engineering and Parallel Computing.

6.1 Motivation

This subsection contextualises the research and pedagogic activities that I aim to pursue at the University of Minho, divided into two topics: Parallel Computing (*PC*); and Software Engineering (*SE*). Initially, my scientific activities will focus on *PC* as it is the field I can contribute the most. However, given my background, professional experience, and the current state-of-the-art within the scientific computing community, both topics are extremely promising research avenues. The potential for cross-pollination between *SE* and *PC* stemmed an emerging area of research that I intend to pursue.

Parallel Computing

Advancements in the performance of microprocessors have slowed down due to limitations in semiconductor technology. The current pace of improvement sits below what Moore's law predicts, which has been reliable since its proposal in 1965 [1], mostly due to transistor leakage and availability of exotic materials used to build transistors. Manufacturers of multicore microprocessors (i.e., *CPUs*) are mitigating the impact of these limitations by increasing the core architecture complexity, implementing operations that simultaneously process batches of data (vector computing), and by packing more cores per chip. For instance, current AMD multicore devices feature up to 64 physical cores with 2-way Simultaneous Multithreading (*SMT*) [2], and current ARM servers have up to 96 cores with 4-way *SMT* and similar vector computing capabilities to AMD and Intel devices [3]. ARM based server-grade solutions are currently being used in industry environments Amazon and Google [4, 5], and with a significant presence on large-scale to exascale scientific clusters, such as the upcoming Minho Advanced Computing Center (*MACC*) Deucalion and BSC MareNostrum5 [6, 7].

A look at the Top500 reveals exponential growth from several gigaflops in 1993 to the first exaflop system in June 2022: the Frontier cluster [1]. The two major challenges to exascale computing have been mitigated, namely energy consumption and fault tolerance, leading to an increase in the proposal and creation of more exascale systems. The authors of [8, 9] have analysed the

1. Access at https://www.top500.org/lists/top500/list/2022/06/.

current state-of-the-art in exascale computing and envision that the two new major roadblocks to *HPC* are software complexity and data volume. The former relates to programming and concurrency models, hardware heterogeneity, scalability, tools, and scientific computing, while the latter concerns the massive amounts of data, and their communication, processed in such systems. The authors identify, in their analysis, the following lines of research as crucial to develop software that adequately use the computational capabilities of exascale systems:

- **Hardware accelerators:** Most Top500 systems use accelerators due to their high computational throughput and energy efficiency. These devices, such as Nvidia's *GPUs* and Google's *TPUs*, have the potential to accelerate calculations that cannot be efficiently performed by multicore devices. However, the fundamental architectural differences between accelerator and multicore devices require different paradigms to design and implement efficient algorithms. There is a need to develop tools and frameworks to ease the development of efficient code for such systems.
- **Workload scheduling:** Improving the traditional programming models (i.e., MPI+OpenMP) and developing alternative models is key. New models should consider the heterogeneity of exascale systems, mainly in the management of data and task allocation and communication to computing devices at the intra- and inter-node levels. Adequate workload scheduling of irregular tasks is crucial for the development of scalable parallel algorithms.
- **Data communication:** Both intra- and inter-node data communication is critical at limiting the scalability of algorithms, due to limited memory and node interconnection bandwidths. This should be considered in the design of scalable algorithms but is also tightly dependent on adequate workload scheduling.
- **Scientific computing:** Computational sciences are one of the major users of petascale and exascale systems [10]. The adaptation of current scientific code to exploit the computational resources of such systems and the development of frameworks that aid non-computer scientists to easily develop efficient code is crucial for the adoption and adequate usage of these complex heterogeneous systems.

While it started as a field accessory to supercomputing and scientific *R&D*, *PC* is an essential part for the advancement of many other fields, such as machine learning (especially artificial intelligence, in which I have co-advised 2 master theses), and scientific computing. The 2019 SPX Workshop Report [9] emphasises the need for collaboration between *PC* and other fields:

"Close collaboration with domain experts outside computer science will [also] be essential to develop domain-specific solutions for challenging emerging applications."

The challenges posed by this type of hardware become even more difficult to overcome by non-computer scientists, a key user group of large-scale systems. Several studies [11, 12, 13] identified causes that lead non-computer scientists to develop inefficient code:

- Focus on problems of their scientific domain leaving little time to tackle code performance.
- Usually lack the expertise to implement, debug, and profile adequate parallelisation approaches.
- Cannot afford to study the architectural details of the target computing devices, limiting the efficiency and portability of their code.

The SPX report also states that irregular workload scheduling will become a non-functional requirement for most applications, as their complexity increases and seek greater concurrency provided by wider computing devices, and an increase in server heterogeneity and in *HPC* systems (from terascale to exascale). Recent features in device design, such as RISC architectures, dynamic power management, varying clock rates, and more complex memory hierarchies also exacerbate the irregularity of workloads. The report outlines four relevant research directions for the future of *PC*, from which these already have been the focus of my scientific career:

- *"(…) amend existing approaches to conversion of scientific theories into computational implementations to handle irregularity more robustly, dynamically balancing computational load and network load across extreme-scale HPC systems."*
- *"(…) new opportunities will come from mating the full system (algorithms-software-hardware) with the computational abstractions of the application at hand (…)"*

My main research focus during my *PhD* has been on improving the performance of scientific code, while merging the gap between the expertise in *PC* and non-computer scientists with *HPC* needs. This resulted in the development of *HEP-Frame*, a framework

with a user-centred development interface that transparently manages the efficient parallel execution of code on multicore and accelerator devices. This work specifically targeted pipelined data analysis applications with highly irregular workloads, and at its core employs a novel multi-layered scheduling approach that automatically manages intra- and inter-node parallelism.

Currently, as a researcher at *MACC*, I am focusing on improving the scalability and performance of both scientific applications and tools of industry partners, while also assessing their expertise and workflows in *PC* in a consultant-like role. Additionally, I am responsible for the development and maintenance of profiling and visualisation tools for massively parallel codes. These research venues already synergise with the *HPC* needs identified in the SPX report.

My scientific contribution in the department research activities should focus in these two topics, developing scheduling strategies for large-scale computing systems, targeting workload distribution and management, and mitigating intra- and inter-node communication costs, with the integration of such advancements in tools for the scientific community and industry. Such tools should be developed in close cooperation with experts from the relevant fields, which allows to study how they can be devised in such a way that *SE* practices are integrated into the computer and non-computer scientist's day-to-day workflows.

Software Engineering

The use of systematic approaches in all phases of software development is crucial to ensure the correctness, robustness, maintenance, and extension capabilities of software. With increasing complexity of applications used in domains that require low fault-tolerance, the adoption of adequate engineering principles from the design to the development and testing of software allows for longer lasting code bases.

SE approaches first appeared in the late 60's within the computer science community, being considered as a pre-requisite to become a software engineer by IEEE since 2005 [14]. However, the adoption by the computational sciences is not yet widespread. The mix of application design and implementation without a clear set of goals is standard among scientific domains that use software development as a research tool. The lack of unstructured code development often leads to ongoing use of legacy software, often low performing, difficult to maintain and extend due to the *"spaghettification"* of its code over the years.

I have been integrated in multi-disciplinary research teams since my master's degree in 2012, specifically among experimental physicists at Laboratório de Instrumentação e Física Experimental de Partículas (*LIP*) [2], and chemical engineers at both *RDB-TS* and Electron Softview, tackling performance issues in legacy code. These experiences exposed me to the struggles of adopting *SE* principles in the workflows of non-computer scientists. While the causes for this behaviour may vary among scientific domains, there are common factors impacting the development of scientific code:

- Scientists are not aware of *SE* principles to produce efficient, robust, modular, and long-lasting code. Few scientists seek training in *SE* [15, 16, 17, 18, 19, 20].
- Functional requirements are usually not known, as development is inherently dependent on the exploratory research in a scientific field. Coding, as the science itself, is approached experimentally. This can cause software packages that grow in popularity to become legacy code [21, 22, 15]. The lack of clear understanding of the requirements has long-lasting implications on the remaining phases of software development.
- Testing and validation is highly dependent on the scientific field, as scientists may not have the observational data to verify the correctness of the implemented scientific model. The lack of systematic testing makes tracing the causes of deviations in results much more difficult, as they may arise either from the algorithm design or implementation [23, 19].
- The use of different terminologies between scientific computing and *SE* aggravates the communication between scientists and software engineers, adding undesirable complexity in multi-disciplinary teams [24, 17].

Scientists rank functional correctness and code performance as top priorities [13]. The mix of functional and non-functional requirements, given an initial poor – or non-existent – design of the software architecture, creates a lack of code structuring between the domain logic and implementation. Additionally, too much focus on performance without adequate know-how can lead to issues in both performance and functional portability, and in software maintainability, if not properly mediated by *SE* practices.

The authors of [20] propose two approaches to bridge the gap between *SE* and scientific computing to mitigate the previously

2. Access at www.lip.pt

raised issues: integrate software engineers in multi-disciplinary teams to develop or refactor existing software; and train non-computer scientists in *SE* practices. The former has been an ongoing practice during my scientific career, where I have co-advised computer science students, integrated into various scientific fields, in the development of tools that meet functional and non-functional requirements of scientists. These tools should have an educational aspect by subtlety persuading users to develop code that follows *SE* practices, either by using Domain Specific Languages [20] or other similar methods, as is the case of *HEP-Frame*.

The training of non-computer scientists in *SE* is currently lacking at the University of Minho. While teaching programming to is common practice, courses such as both Licenciatura and Mestrado em Engenharia Física do not offer a curricular unit in *SE*[3] applied to the needs of scientific computing. It is crucial to teach and show how *SE* principles can be integrated in the workflows of scientists if these practices are to be generally adopted.

6.2 Scientific Program - DC5

This section presents my scientific contribution to the department research activities that could be adequately explored in two complementary and relevant fields: scheduling of irregular workloads for heterogeneous systems in large-scale computing systems; and development of efficient tools for the scientific and industry users of these systems. While a significant part of my research has been related to these topics, I want to further extend the work done so far and tackle the issues posed by emerging hardware solutions for real-world scientific and industry software.

Irregular Workload Scheduling

I have tackled the challenges of scheduling the execution of irregular tasks and data among the resources of multi-socket servers with accelerators since my *PhD*. One outcome of this research was a novel multi-layer scheduler to adaptively handle parallel I/O operations and computations during runtime, targeted at pipelined data analysis applications. It dynamically allocates different resources depending on the computational needs of the data analysis (either I/O-, memory-, or compute-bound), with no prior knowledge of the characteristics of tasks, input data, and hardware. It was also capable of offloading specific computations to *GPUs*, while mitigating data communication costs through between multicore and accelerator devices.

The 2019 SPX Workshop Report [9] underlines the importance of such approach given the current advancements in both hardware and domain-specific software:

"Given the conflict between the simple models desired and needed by the theory community, and the complication of real hardware, another research direction is to develop a multi-layered approach to algorithm design and analysis with a succession of more refined models, each capturing more detail of the hardware and better predicting runtime or other resource usage."

This approach contrasts with the standard of developing efficient algorithms targeted at a wider range of applications. Their increased abstraction inherently neglects most computational characteristics of the algorithms, reducing the potential for performance improvements. I believe that my expertise on these issues would benefit the department scientific research on massively parallel systems:

- Vectorisation aware scheduling.
- Irregular workload distribution and monitoring in large-scale computing systems.
- Mitigation of intra- and inter-node communication overhead.

Vector operations are fundamental to use the full performance of multicore and many accelerator devices. While 512-bit vector registers are available in Intel and ARM devices, with several vector-capable *ALUs* per core, wider vectors and more *ALUs* are expected to become standard. Automatic code vectorisation in parallel environments is not enough, as multiple threads/processes can stress shared caches and lead to delays in loading data from memory to vector registers, limiting the possible performance improvements. Schedulers often neglect the vector computing capabilities of recent architectures. I intend to explore strategies for better cache management in heavily vectorised parallel code, designing schedulers that consider per-core vector usage to better balance irregular workloads.

3. https://www.uminho.pt/PT/ensino/oferta-educativa/Cursos-Conferentes-a-Grau/_layouts/15/UMinho.PortalUM.UI/Pages/CatalogoCursoDetail.aspx?itemId=4104&catId=12
https://www.uminho.pt/PT/ensino/oferta-educativa/Cursos-Conferentes-a-Grau/_layouts/15/UMinho.PortalUM.UI/Pages/CatalogoCursoDetail.aspx?itemId=4102&catId=12

The adequate scheduling of irregular workloads is crucial to exploit the computational resources in heterogeneous servers and large-scale systems. While the scheduling between multicore and accelerator devices is still a relevant area of research, I intend to use the expertise in intra-node scheduling I gathered during my *PhD* to focus on inter-node workload management. The heterogeneity of servers in large-scale systems poses challenges in performance modelling of the hardware, varying server throughput (due to different hardware and/or energy requirements), and application dataset-dependent behaviour, assuming no prior knowledge about the workloads to be managed. Additionally, research on speculative scheduling and execution of inter-dependent tasks, which is a common pattern in scientific code, is still in its infancy. I am already assessing the feasibility of incorporating speculative execution in task schedulers through the orientation of a master's thesis, and I intend to further pursue this topic.

Data communication is one of the major limiting factors to scale application performance to hundreds or thousands of computing devices, either inside the same server or across multiple interconnected nodes. These issues should be considered in the design of highly parallel algorithms, by exploiting the complex hardware-specific memory hierarchies, but also on the scheduling of workloads among devices, minimising synchronisations, and the impact of data transfers across interconnected nodes. Although these two challenges may require different approaches to be tackled, both are deeply related in the design of workload schedulers for massively parallel systems. I intend to apply my expertise to improve data throughput in vectorisable algorithms and mitigate communication costs among many interconnected heterogeneous nodes in large-scale systems, which are critical limitations to performance that I identified across several application domains during my time at *MACC*.

Domain Specific Tools for Large-scale Computing Systems

The proposed research directions synergise with the development of tools to aid the development of high-performance code. Such tools, either standalone or frameworks/libraries, should be built on strong software engineering foundations to ensure their robustness and longevity, with a clear definition of their scope in the context of the requirements of individual *HPC* domains. While the research of scheduling strategies is relevant on its own, I believe that such advancements should be integrated into domain specific software that can have an immediate impact on the work of others.

Frameworks also have the potential to persuade its users to the adoption of software engineering principles by constraining the code organisation, data structuring, and overall design of the architecture of applications. If devised and applied correctly, these constraints lead to users defining the application requirements and thinking about the code structure and its execution flow before developing their code, resulting in more robust and longer lasting code bases. I have already applied this strategy on *HEP-Frame* with very promising results, as analysed in my *PhD* dissertation.

I have identified two short-term goals in the development of domain specific tools:

- **Monitoring Task Execution Graphs (TEGs):** During my research I found that there is a lack of tools for the visual analysis of graphs, displaying information relevant for a performance assessment. The feasibility of developing a tool to visualise, validate, and debug the structure of complex *TEGs*, often at the core of irregular workload scheduler's logic, was assessed and resulted in a proof-of-concept [4].
- **Frameworks for data analysis applications:** The exploration of the novel scheduling strategies should be paired with the development of tools for their integration into real workflows on large scale systems. The implementation of schedulers in such frameworks benefit their development through the feedback of real users, while having an immediate positive impact on scientific and industry research.

Additionally, I will focus on obtaining funding through research projects and industry partners. I aim to leverage my multi-disciplinary collaborations in the short-term for the submission of joint proposals for EuroHPC and the National Network for Advanced Computing (*RNCA*) funded projects, focused on the advancement of the topics proposed in this project. Industry-wise, I intend to leverage my connection with Electron Softview and *MACC* partners. I am currently assessing this type of collaborations through the supervision of a master thesis with Electron Softview. If successful, these exploratory projects will lead to closer cooperation and funding opportunities directly through the companies and/or through joint proposals to European projects.

4. Further information available at https://ampereira90.github.io/portfolio/

6.3 Pedagogic Program - CP7

PC is often viewed a field targeted at the exploitation of exotic computing resources used by specific types of applications. This misconception diminishes the importance of *PC* in the global context of computer science, especially in the formative steps of software engineers, since top end hardware is rarely used outside of *R&D*. However, the architectural characteristics of such hardware eventually become common. While it is now usual to find multicore devices on laptops, desktops, and even smartphones with at least 8 cores, a decade ago those were only available in cutting-edge computing clusters. Since the advancements in computer hardware often appear in these specialised environments, they should be used as training tools to prepare software engineers for the mainstream hardware of the future.

The University of Minho is at the forefront of *PC* infrastructure in Portugal as the host institution of *MACC* [5] and *SeARCH* [6], which offer state-of-the-art resources, such as the new ARM based Fujitsu multicore devices, Intel and AMD devices, and several hardware accelerators. It is crucial that the university offers the best training possible to prepare the next generation of scientists to leverage these resources.

A portion of my pedagogic activities focus on training two communities that benefit from *PC*: software engineers and non-computer scientists. Different approaches are required to meet distinct goals:

- Software engineers should have the expertise to understand the architecture of exotic devices. These skills can be used to analyse the next generation of computing devices and consequently develop efficient code. For instance, ARM based multicore devices have been available in computing clusters, such as *SeARCH*, but no training is offered for all software engineers. With this technology now mainstream, few developers have the expertise to produce efficient code. Universities should not react but prepare their curriculums to these emerging technologies.
- Non-computer scientists develop and use software as tools for their research, having reduced expertise in computer science. Training in *PC* should be complemented with the *SE* principles useful for the development of scientific software, such as the requirement analysis of features and performance, the use of adequate design patterns for parallel algorithms, and code development workflows. These scientists should be provided with the necessary training to develop code that is not only efficient, but also functionally correct and easy to maintain.

The 2021/22 year was the first with Programação Paralela as a mandatory curricular unit in the training of students that undergo the Mestrado em Engenharia Informática and Licenciatura em Engenharia Física at the University of Minho. Its current approach to lab classes should be improved by shifting the focus to newer architectures, reflecting the current market, targeting parallel computing for ARM devices and large-scale systems. I aim to continue integrating these students into both academic and industry *R&D* environments through the orientation of master and doctoral theses.

Scientific Computing with Software Engineering

While the course on Programação Paralela for the Licenciatura em Engenharia Física is targeted at the computing needs of non-computer and data scientists, I believe that it should be complemented with training in specific *SE* methodologies, which is currently lacking. Similar courses are already available, such as CS294-73 at the University of California [7] and at *TACC* [8], reinforcing the credibility of this type of training for scientific fields that resort to software development. Ideally, programming models, workload scheduling, and communication optimisations for exascale computing should also be taught, but those subjects require expertise that cannot be expected at the BSc level.

This subsection presents a summarised proposal for a curricular unit entitled “Performance Engineering for Scientific Computing”, which bridges the gap between *PC* and *SE* oriented at non-computer scientists with basic programming experience.

Course Description

This course focus on the development of skills in the design, implementation, and parallelisation of efficient code in the context of popular algorithmic patterns in scientific computing [10]. It is suitable for 1h+3h or 2h+2h theoretical/lab classes per week.

5. Available at https://macc.fccn.pt/
6. Available at http://search6.di.uminho.pt/wordpress/
7. Available at https://inst.eecs.berkeley.edu/ cs294-73/
8. Available at https://www.tacc.utexas.edu/education/undergrads-grads/academic-courses

Theoretical classes with no mandatory attendance and all resources (slides and video recording) available after the class for the students to access. Each lab class is accompanied by a "lab guide", which includes a simple homework section with the purpose of preparing the computing environment and case studies at home. Lab class attendance is mandatory to ensure the authenticity of the submitted homeworks [9].

Goals and Learning Outcomes

The students should develop skills in:

- The analysis of algorithms behaviour and complexity to identify parallelisation opportunities.
- Parallelisation of algorithms (multithreading vs multiprocess).
- Understanding and using the algorithm design patterns most common in scientific computing.
- Design of data structures adequate to algorithm patterns and performance requirements.
- Analysis and characterisation of the computational behaviour of parallel algorithms in terms of efficiency, scalability, and bottlenecks.
- *SE* principles for the development of large-scale simulations.

Pre-requisites

Basic knowledge of imperative programming languages (C/C++ preferred) and the hardware and software structures of computing systems. Basic understanding of UNIX based systems. Students should belong to a domain that uses complex scientific simulations, such as experimental physics, biology, chemistry, or data sciences.

Program

1. Algorithmic patterns in scientific computing
   - Structured and unstructured grids;
   - Fast Fourier Transforms (**Book3**: 12.2, 13.9);
   - Monte Carlo methods (**Book3**: 7.7, 7.9);
   - Sparse and dense matrices (**Book2**: 4.12, 5.14; **Book3**: 2.7).
2. Code parallelisation and profiling
   - Vectorisation (**Book1**: B.2, 4.2, 4.3, 4.5; **Book2**: 6.3);
   - Multithread vs multiprocess parallelisation with OpenMP and MPI (**Book1**: 3.11, 5; **Book2**: 6.7, 6.8; **Book4**: 4.4 - 4.7, 5.4 – 5.6);
   - Introduction to profiling tools (Intel VTune and Advisor);
   - Identification and characterisation of performance bottlenecks (**Book1**: 5.2; **Book4**: C);
   - Parallelisation scalability analysis (**Book1**: 6.2).
3. Software design and management
   - Brief introduction to requirement analysis;
   - Parallelisation strategies: the *PCAM* methodology (**Book4**: 2.2);
   - Task and data decomposition patterns (**Book4**: 2.3);
   - Parallelisation design patterns (**Book4**: 2.4, 2.5);
   - Revision control with Git;
   - Introduction to unit testing (**Book5**: 3, 4).

Grading

Three components contribute for the final grade:

- Optional homework assignments (10% of the final grade).

9. This approach has already been used and refined in the curricular units I currently teach with improved productivity in the lab classes and positive student feedback. Similar approaches are gaining widespread adoption in Portugal according to Semanário Expresso (number 2540, $\mathbf{2}^{nd}$ of July 2021): https://github.com/ampereira90/ampereira90.github.io/blob/master/files/artigo_expresso.pdf.

- Written test to evaluate cognitive and intellectual capabilities (40%).
- Final group project (50%)
- Development, parallelisation, and analysis of a small-scale scientific application.
- Deliver a short report and presentation on the development and computational analysis of the algorithms implemented.

Bibliography

- **Book1**: David Patterson and John Hennessy, *Computer Architecture. A Quantitative Approach*, 6^{th} Edition, Morgan Kaufmann, 2017.
- **Book2**: David Patterson and John Hennessy, *Computer Organization and Design*, 5^{th} Edition, Morgan Kaufmann, 2013.
- **Book3**: William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, 3^{rd} Edition, Cambridge University Press, 2007.
- **Book4**: Gerassimos Barlas, *Multicore and GPU Programming: an Integrated Approach*, 1^{st} Edition, Morgan Kaufmann, 2015.
- **Book5**: Jeff Langr, *Modern C++ Programming with Test-Driven Development*, 1^{st} Edition, The Pragmatic Bookshelf, 2014.

6.4 References

[1] G. E. Moore, "Cramming More Components onto Integrated Circuits," *Proceedings of the IEEE*, vol. 86, no. 1, pp. 82–85, 1998.

[2] AMD, "AMD Epyc™ 7003 Series Processors the New Standard for the Modern Data Center," 2021, Technical Report. [Online]. Available: https://www.amd.com/system/files/documents/amd-epyc-7003-series-datasheet.pdf

[3] A. Frumusanu, "Marvell Announces ThunderX3: 96 cores & 384 Thread 3rd Gen ARM Server Processor," 2020, accessed in 25^{th} July 2022. [Online]. Available: https://www.anandtech.com/show/15621/marvell-announces-thunderx3-96-cores-384-thread-3rd-gen-arm-server-processor

[4] R. Mathá, D. Kimovski, A. Zabrovskiy, C. Timmerer, and R. Prodan, "Where to Encode: A Performance Analysis of x86 and Arm-based Amazon EC2 Instances," in *2021 IEEE 17th International Conference on eScience (eScience)*, 2021, pp. 118–127.

[5] A. Vahdat, "The Past, Present and Future of Custom Compute at Google," 2021, accessed in 25^{th} July 2022. [Online]. Available: https://cloud.google.com/blog/topics/systems/the-past-present-and-future-of-custom-compute-at-google

[6] MACC, "MACC Resources," 2022, accessed in 26^{th} July 2022. [Online]. Available: https://macc.fccn.pt/resources

[7] T. Morgan, "ATOS Wins MareNostrum 5 Deal at Barcelona Supercomputing Center," 2022, accessed in 24^{th} July 2022. [Online]. Available: https://www.nextplatform.com/2022/06/16/atos-wins-marenostrum-5-deal-at-barcelona-supercomputing-center/

[8] J. Shalf, S. Dosanjh, and J. Morrison, "Exascale Computing Technology Challenges," in *High Performance Computing for Computational Science – VECPAR 2010*. Springer Berlin Heidelberg, 2011, pp. 1–25.

[9] S. D. Stoller, M. Carbin, S. Adve, K. Agrawal, G. Blelloch, D. Stanzione, K. Yelick, and M. Zaharia, "Future Directions for Parallel and Distributed Computing: SPX 2019 Workshop Report," in *SPX Workshop Reports*, 2019.

[10] M. D. Jones, J. P. White, M. Innus, R. L. DeLeon, N. Simakov, J. T. Palmer, S. M. Gallo, T. R. Furlani, M. Showerman, R. Brunner, A. Kot, G. Bauer, B. Bode, J. Enos, and W. Kramer, "Workload Analysis of Blue Waters," 2017. [Online]. Available: https://arxiv.org/abs/1703.00924

[11] J. E. Hannay, C. MacLeod, J. Singer, H. P. Langtangen, D. Pfahl, and G. Wilson, "How do Scientists Develop and Use Scientific software?" in *2009 ICSE Workshop on Software Engineering for Computational Science and Engineering*, 2009, pp. 1–8.

[12] P. Prabhu, T. B. Jablin, A. Raman, Y. Zhang, J. Huang, H. Kim, N. P. Johnson, F. Liu, S. Ghosh, S. Beard, T. Oh, M. Zoufaly, D. Walker, and D. I. August, "A Survey of the Practice of Computational Science," in *State of the Practice Reports*, ser. SC '11. Association for Computing Machinery, 2011.

[13] J. C. Carver, R. P. Kendall, S. E. Squires, and D. E. Post, "Software Development Environments for Scientific and Engineering Software: A Series of Case Studies," in *29th International Conference on Software Engineering (ICSE'07)*, 2007, pp. 550–559.

[14] P. Bourque and R. Fairley, "Guide to the Software Engineering Body of Knowledge," 2014, iEEE. [Online]. Available: www.swebok.org

[15] V. R. Basili, J. C. Carver, D. Cruzes, L. M. Hochstein, J. K. Hollingsworth, F. Shull, and M. V. Zelkowitz, "Understanding the High-Performance-Computing Community: A Software Engineer's Perspective," *IEEE Software*, vol. 25, no. 4, pp. 29–36, 2008.

[16] J. Carver, D. Heaton, L. Hochstein, and R. Bartlett, "Self-Perceptions about Software Engineering: A Survey of Scientists and Engineers," *Computing in Science & Engineering*, vol. 15, no. 1, pp. 7–11, 2013.

[17] S. M. Easterbrook and T. C. Johns, "Engineering the Software for Understanding Climate Change," *Computing in Science & Engineering*, vol. 11, no. 6, pp. 65–74, 2009.

[18] R. Sanders and D. Kelly, "Dealing with Risk in Scientific Software Development," *IEEE Software*, vol. 25, no. 4, pp. 21–28, 2008.

[19] J. Segal, "Some Problems of Professional End User Developers," in *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2007)*, 2007, pp. 111–118.

[20] A. Johanson and W. Hasselbring, "Software Engineering for Computational Science: Past, Present, Future," *Computing in Science & Engineering*, vol. 20, no. 02, pp. 90–109, 2018.

[21] J. Segal, "When Software Engineers Met Research Scientists: A Case Study," *Empirical Software Engineering*, vol. 10, no. 4, pp. 517–536, 2005.

[22] L. Hochstein, J. Carver, F. Shull, S. Asgari, V. Basili, J. K. Hollingsworth, and M. V. Zelkowitz, "Parallel Programmer Productivity: A Case Study of Novice Parallel Programmers," in *SC'05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*. IEEE, 2005, pp. 35–35.

[23] F. Shull, J. Carver, L. Hochstein, and V. Basili, "Empirical Study Design in the Area of High-Performance Computing (HPC)," in *2005 International Symposium on Empirical Software Engineering, 2005.* IEEE, 2005, pp. 10–20.

[24] S. Faulk, E. Loh, M. L. V. D. Vanter, S. Squires, and L. G. Votta, "Scientific Computing's Productivity Gridlock: How Software Engineering Can Help," *Computing in Science & Engineering*, vol. 11, no. 6, pp. 30–39, 2009.

Other Activities

7.1 Coordination and Involvement in Technological Specialisation Courses - OAR1

Responsible for the restructuring of the *Arquitetura de Dispositivos Móveis* (ADM, in 2015/2016) curricular unit of the *Desenvolvimento de Aplicações Móveis* professional degree, at *Instituto Politécnico do Cávado e Ave*. The previous and first edition of the *ADM* was not properly organised according to the overall goals of the course it was integrated in. I was invited to lecture and re-structure this curricular unit given my background in teaching computer architecture and computing systems. I was in charge of:

- Definition of the learning goals, expected learning outcomes, and evaluation methodology.
- Planing the concepts to be approached in each session with the students.
- Creation of content for theoretical classes (slides and other support material).
- Creation of the content for the lab classes, together with another invited teaching assistant (Marco Couto).

7.2 Scientific Dissemination - OAR2

May 2022 — Presenter of a 20-min session on the engineering challenges tackled at the Electron Softview Spin-Off in "Tomorrow Needs Engineering - da Academia ao Mercado", organised by InvestBraga at the Altice Forum in Braga, Portugal. This session was integrated in the 5^a *Semana da Economia* and focused on the application of academia research to industry needs.

March 2022 — Lecturer of a 3-hour workshop entitled "Programming Microcontrollers: A Computer Scientist's Perspective", invited by the Núcleo de Estudantes de Física da Universidade do Minho, targeted at physics students. The theoretical and hands-on content, created specifically for this workshop, is available at https://am-pereira90.github.io/teaching/.

June 2016 — Teaching assistant at the 2016 Summer School in Advanced Scientific Computing, organised by the Texas Advanced Computing Center (*TACC*) at the University of Minho. This 3-day school focused on creating the theoretical and practical foundations for the development of efficient parallel code targeted at the non-computing scientific community. The goal was to give the tools and expertise to scientists from the most diverse fields to efficiently use the university's advanced computing resources, such as the *SeARCH* computing cluster.

7.3 Jury of Academic Degrees - OAR4

Participation in an academic jury as external evaluator:

October 2021 "Optimizing Real-time Physics Simulation using *GPGPU* Techniques", co-supervised by Professor Jorge Barbosa and Nelson Rodrigues and successfully defended by Ricardo Pereira, for the degree in *Mestrado Integrado em Engenharia Informática e Computação* at the *Faculdade de Engenharia da Universidade do Porto*.

Universidade do Minho
Escola de Engenharia

Certificate of Attendance

Summer School in Advanced Scientific Computing

20th - 23th June 2016, Braga Portugal

André Pereira

Chair of Summer School

UT Austin | Portugal

International Collaboratory for Emerging Technologies, CoLab

Certificate of Presentation

This is to certify that

Andre Pereira

has presented the paper entitled

HEP - Frame: A Software Engineered Framework to Aid the Development and Efficient Multicore Execution of Scientific Code

at The 2015 International Conference on Computational Science
&
Computational Intelligence (CSCI'15)

December 7-9, 2015 | Las Vegas, USA

Kaveh Arbtan on behalf of Steering Committee

CSCI'15
December 07-09, 2015
Monte Carlo Resort
Las Vegas Nevada, USA

European Organization for Nuclear Research
Organisation européenne pour la recherche nucléaire

CERN School of Computing
CERN – IT Department
CH – 1211 Genève 23

Tel direct: + 41 22 767 5548
Fax: + 41 22 766 9941
computing.school@cern.ch

Our reference/notre référence : iCSC2015-yh

Geneva, 26th February 2015

TO WHOM IT MAY CONCERN

I certify that

André MARTINS PEREIRA

attended the 2015 inverted CERN School of Computing as lecturer.

The school took place at CERN, Geneva, Switzerland on 23-24 February 2015.

Yasemin Hauser

Administrator
CERN School of Computing

www.cern.ch

Braga, Portugal
25 August to 05 September 2014

CERN School of Computing Diploma

I, undersigned, certify that

PEREIRA André

has successfully passed the final examination.

The programme consisted of 50 hours of lectures and hands-on exercises.

The 2014 CERN School of Computing was jointly organized by the European Organization for Nuclear Research (CERN), Geneva, Switzerland and the University of Minho and LIP, Braga, Portugal

Alberto Pace
Director
CERN School of Computing
05 September 2014

Certificate of Attendance

This is to certify that

André Pereira

has attended the

Summer School on Paralell High Performance Computing using Accelerators

held at Universidade do Minho in Braga (Portugal) from the 25th to the 27th of June 2014

Professor Alberto José Proença
Summer School Director

MAP i DOCTORAL PROGRAMME IN COMPUTER SCIENCE

Certificate of Attendance

This is to certify that

André Martins Pereira

attended the

MAP-i Spring School on Logic of Dynamical Systems

held at the University of Minho in Braga (Portugal) from the 24th to the 28th of March 2014 under the coordination of Prof. André Platzer (Logical Systems Lab, Carnegie Mellon University, USA).

Prof. Luís Soares Barbosa
MAP-i Director

4TH IDPASC SCHOOL

Braga, 20 - 28 January 2014

To whom it may concern

We certify that

Pereira, André

has attended the "**Fourth IDPASC School Braga - 2014**", wich was held at Campus de Gualtar - Universidade do Minho, Braga, Portugal from Braga, 20-28 January 2014

Braga, 20 January 2014
The Organizing Committee

Laboratório de Instrumentação e Física Experimental de Partículas
Av. Elias Garcia 14-1º - 1000-149 Lisboa - Portugal
NIF: 501 694 650

The IC^2 Institute
at

THE UNIVERSITY OF TEXAS AT AUSTIN
presents this certificate of accomplishment to

Andre Martins Pereira

In recognition of completing

2012 Summer Internship in Advanced Computing Research

July 10th - August 12th, 2012

Dr. Keshav Pingali, Chair of Distributed & Grid Computing
Director, UT Austin CoLab Advanced Computing Program

GOES 463

LITHO IN U.S.A.

Universidade do Minho
Serviços Académicos

Carla Isabel Pereira Lavrador, Registrar of the University of Minho, hereby certifies, according to University records, that André Martins Pereira, born in the parish of União das freguesias de Viana do Castelo (Santa Maria Maior e Monserrate) e Meadela, in the county of Viana do Castelo, in the district of Viana do Castelo, son of Jaime de Amorim Pereira and of Elisabete Amorim Martins Pereira, obtained the following passing grades in the Doctoral degree in Computer Science, under the joint doctoral programme with the Universities of Aveiro, Minho and Porto:

Option in Computation Paradigms – 16 (sixteen), through equivalency – 5 ECTS;
Option in Theory and Fundamentals – 14 (fourteen), through equivalency – 5 ECTS;
Seminar – 19 (nineteen), through equivalency – 15 ECTS;
Option in Technology – 17 (seventeen), through equivalency – 5 ECTS;
External Option – External Option – 19 (nineteen), through equivalency – 5 ECTS;
Free Option – Free Option – 18 (eighteen), through equivalency – 5 ECTS;
Preparation of Thesis Proposal – 19 (nineteen), through equivalency – 20 ECTS.

She also certifies that he completed the above-mentioned Degree on the 27th of May, 2019. Thereupon, the members of the Doctoral Examining Committee, expressly appointed for the occasion, have met on the said date, and decided to confer the Doctor degree to the candidate – Approved, with the final grade of Very Good.

This same certificate carries the embossed seal of this University.

The Registry of the University of Minho, on 04th of July, 2019.

The Registrar,

universidade de aveiro

Universidade do Minho

U.PORTO