

Marco Cicalè Baztán

Ph.D. student, Universidad Politécnica de Madrid (UPM) & IMDEA Software Institute
mciccaleb@gmail.com • 0009-0000-8821-0587 • <https://mciccale.github.io/>

Research statement

My central research interest is applying formal, rigorous mathematical techniques throughout the entire software production lifecycle. In particular, I am interested in the theory and application of static analysis, verification, programming languages implementation, formal semantics, logic, and order theory for the design, implementation, and deployment of systems. Currently, my work is the design and implementation of tools and techniques for the (static) analysis and (compile-/run-time) verification of declarative languages, along three axes: *expressiveness*, handling richer language constructs and properties; *efficiency*, reducing the overhead of fully safe software systems; and, *explainability*, making static analysis progress and results transparent and understandable to the user.

Education

Ph.D. in computer science

2025–present

SCHOOL: Universidad Politécnica de Madrid (UPM) & IMDEA Software Institute, Madrid, Spain.

SUPERVISORS: José F. Morales, Manuel V. Hermenegildo.

TOPIC: Techniques and tools for static analysis by top-down abstract interpretation of programs with assertions in the constrained Horn clauses intermediate representation. The goal is to make such analysis more *expressive* (e.g., supporting higher-order constructs), more *efficient* (e.g., abstract multivariance for reducing run-time checks), and more *explainable* (e.g., a source-level abstract debugger for the PLAI/TD solver).

M.Sc. in formal methods in computer science and engineering

2024–2025

SCHOOL: Universidad Politécnica de Madrid (UPM) & Universidad Complutense de Madrid (UCM), Madrid, Spain.

COURSES: Theory of Programming Languages (formal semantics, type systems), Static Analysis and Constraint Solving Problems (abstract interpretation, CDCL/DPLL SAT/SMT solvers), Computer-Aided Program Verification (Hoare logic, Dafny, Liquid Haskell).

THESIS: *Towards Verification of Higher-Order (Constraint) Logic Programs via Abstract Interpretation* — 10/10 with honours.

GRADE: 9.2/10

B.Sc. in computer engineering

2020–2024

SCHOOL: Universidad Politécnica de Madrid (UPM).

THESIS: *Improvements in Ciao Prolog's Development Environment: Developing A Rich Development Environment for Ciao on Visual Studio Code* — 10/10 with honours.

GRADE: 7.4/10

Publications

- [1] D. Ferreira, D. Jurjo-Rivas, **M. Cicalè**, J.F. Morales, P. López-García, M.V. Hermenegildo.
Exploiting Multiple Abstract Call Patterns for Optimizing Run-Time Checks.
To appear in *Theory and Practice of Logic Programming* (TPLP), special issue for the *International Conference of Logic Programming* (ICLP), 2026.
[doi:10.48550/arXiv.2606.01076](https://doi.org/10.48550/arXiv.2606.01076)
- [2] **M. Cicalè**, D. Jurjo-Rivas, J.F. Morales, P. López-García, M.V. Hermenegildo.
Hiord[‡]: An Approach to the Specification and Verification of Higher-Order (C)LP Programs.
In *Theory and Practice of Logic Programming* (TPLP), special issue for the *International Conference of Logic Programming* (ICLP), 2025.
[doi:10.1017/S147106842510015X](https://doi.org/10.1017/S147106842510015X)

Systems

Ciao system

<https://ciao-lang.org/>

Modern, multi-paradigm programming system with an advanced programming environment. The main motivation behind the system is the integration of an expressive and extensible programming language, and a powerful and efficient program preprocessor that helps programmers produce in less time and with less effort code that has fewer or no bugs. The preprocessor is built around an abstract interpretation-based static analyzer capable of inferring program (hyper)properties such as types/shapes, modes, determinacy, cost bounds, discharging as many checks as possible at compile time and generating run-time checks only when necessary. As a member of the Ciao development team, I contribute to the maintenance of the top-down abstract interpretation framework described above, and to the implementation of our novel tools and techniques.

Skills

- **Programming languages:** Prolog, Elixir, Python, Java, Lisp/Elisp, JavaScript, TypeScript.
- **Formal methods:** Type/Shape Analysis, Sharing Analysis, Abstract Interpretation, Formal Semantics, (Program) Logics, Constrained Horn Clauses, Deductive Verification, SAT/SMT solving.
- **Tools:** Emacs, Linux, Git, Bash, $\text{\LaTeX}$.
- **Spoken/Written languages:** Spanish (native), English (fluent).

Research experience

Research intern

2023–2025

INSTITUTION: IMDEA Software Institute, Madrid, Spain.

SUPERVISORS: José F. Morales, Manuel V. Hermenegildo, Pedro López-García.

TOPIC: Design and implementation of a static analysis for higher-order (C)LP programs; led to a journal publication [2].

Talks

- *Exploiting Multiple Abstract Call Patterns for Optimizing Run-Time Checks.*
ICLP'26 (FLoC'26), Lisbon, Portugal; July 2026.
- *Hiord[#]: An Approach to the Specification and Verification of Higher-Order (C)LP Programs.*
ICLP'25, Rende (CS), Italy; September 2025.

Service

Sub-reviewer: International Conference of Logic Programming (ICLP), 2025.

Dissemination

AdaByron programming contest [\(post\)](#)

July 2026

Disseminated, motivated, and encouraged participants of the national *AdaByron* programming contest to explore and get to know *formal methods* and its impact in society, through an outreach stand at the venue.

Madrid es Ciencia fair [\(post\)](#)

March 2026

Volunteered at IMDEA Software Institute's stand at the venue, introducing high-schoolers to key concepts in computer science and the importance of reliable software through games.

Software Matters gymkhana [\(post\)](#)

November 2025

Volunteered in the fifth edition of the gymkhana. Prepared a logical thinking activity for high-school students and guided them through the process of solving the problem using propositional logic.