Mickaël Laurent

Postdoctoral Researcher in Computer Science

mlaurent@irif.fr
mlaurent.ovh
https://github.com/E-Sh4rk
https://orcid.org/0000-0003-1590-2392

Education

2020 – 2024 **PhD of Computer Science**, *Institut de Recherche en Informatique Fondamentale (IRIF)*, Paris (France)
Polymorphic type inference for dynamic languages: reconstructing types for systems combining parametric, ad-hoc, and subtyping polymorphism. Supervisors: Giuseppe Castagna & Kim Nguyen.

2016 – 2020 **Graduation and Master degree**, *Ecole Normale Supérieure Paris-Saclay*, Cachan (France)
Graduated in computer science in 2017, with high honors.
Master degree obtained in 2019: Algorithm and Foundations of Programming (MPRI), with high honors.
Master degree obtained in 2020: Diploma of Ecole Normale Supérieure Paris-Saclay.
Main courses: algorithmics, complexity, logics, automata theory, programming languages, type systems.

2014 – 2016 **Preparatory classes for the Grandes Ecoles**, *Lycée Champollion*, Grenoble (France)
MPSI/MP*. Main subjects: Math, Physics, Computer Science, English, Philosophy.

2014 **Baccalaureate S option Engineering Science**, *Lycée Paul-Héroult*, Saint-Jean de Maurienne (France)

Experience

2024 – Now **Postdoctoral researcher position**, *Charles University*, Prague (Czech Republic)
Supervisor: Jan Vitek.

2023 – 2024 **Temporary Teaching and Research Assistant (ATER)**, *Université Paris-Cité*, Paris (France)
Teaching of practical activities and tutorial groups to classes of ∼35 undergraduate students (168h).

Summer 2022 **Orgeon Programming Languages Summer School**, *University of Oregon*, Eugene (USA)
Participation to the OPLSS. During 2 weeks, lectures were given by experts in the programming language area.

October 2019 – July 2020 **SAT-based model-checking (internship, 10 months)**, *University of Iowa*, Iowa City (USA)
Implementation of methods aiming to improve the compositional safety analysis within the SAT-based model-checker Kind2: computation of Minimal Inductive Validity Cores, Minimal Cut Sets, etc. Supervisor: Cesare Tinelli.

March 2019 – July 2019 **Set-theoretic type systems (internship, 20 weeks)**, *LRI, Université Paris-Saclay*, Paris (France)
Formalization of type inference for a language with subtyping and intersection types. Study of some possible extensions for gradual typing and polymorphism. Supervisors: Kim Nguyen and Giuseppe Castagna.

March 2018 – July 2018 **Invariant synthesis (internship, 5 months)**, *Cylab, Carnegie Mellon University*, Pittsburgh
Automatic synthesis of inductive invariants (in the Bernays-Schönfinkel fragment of the first-order logics) for the Ivy language, using technics of counterexample generalization. Supervisor: Bryan Parno.

Summer 2017 **Static analysis of models (internship, 8 weeks)**, *Carnegie Mellon University*, Pittsburgh (USA)
Counterfactual causal analysis of Kappa models. Kappa is a rule-based language used for modelling cellular signaling. My goal was to generate causality graphs regrouping events from different simulations. Supervisor: Jean Yang.

Publications

January 2024 **Polymorphic Type Inference for Dynamic Languages**, *POPL 2024*, conference paper
Giuseppe Castagna, Mickaël Laurent, and Kim Nguyen. 2024. Polymorphic Type Inference for Dynamic Languages. Proceedings of the ACM on Programming Languages 8, POPL (January 2024).
https://doi.org/10.1145/3632882

January 2024 **Prototype of Typechecker**, *Artifact Digital Object Group*, software artifact
Giuseppe Castagna, Mickaël Laurent, and Kim Nguyen 2024. Prototype Typechecker for the Article "Polymorphic Type Inference for Dynamic Languages." Association for Computing Machinery (ACM).
https://doi.org/10.5281/zenodo.8408276

January 2022 **On Type-Cases, Union Elimination and Occurrence Typing**, *POPL 2022*, conference paper
Giuseppe Castagna, Mickaël Laurent, Kim Nguyen, and Matthew Lutze. 2022. On type-cases, union elimination, and occurrence typing. Proceedings of the ACM on Programming Languages 6, POPL (January 2022).
https://doi.org/10.1145/3498674

May 2022 **Revisiting occurrence typing**, *Science of Computer Programming*, journal article
Giuseppe Castagna, Victor Lanvin, Mickaël Laurent, and Kim Nguyen. 2022. Revisiting occurrence typing. Science of Computer Programming 217, (May 2022), 102781. https://doi.org/10.1016/j.scico.2022.102781
We revisit occurrence typing, a technique to refine the type of variables occurring in type-cases.

August 2021 **Merit and Blame Assignment with Kind 2**, *Formal Methods for Industrial Critical Systems*
Daniel Larraz, Mickaël Laurent, and Cesare Tinelli. 2021. Merit and Blame Assignment with Kind 2. In Formal Methods for Industrial Critical Systems. Springer International Publishing, 212–220.
https://doi.org/10.1007/978-3-030-85248-1_14