–PhDs, Postdocs, internships, permanent and temporary research positions–

–Formal methods, Safety, and Artificial Intelligence–

Keywords: formal methods, artificial intelligence, safety, machine learning, SMT solving, abstract interpretation, static analysis, symbolic execution, software engineering

Programming Languages: *OCaml*, *Python*

Tools: *Frama-C*, *PyTorch*, *TensorFlow*, Keras

Context: CEA LIST, Software Security Lab

The Software Reliability Laboratory (LSL) at CEA LIST has an ambitious goal: help designers, developers and validation experts produce high-confidence systems and software. As our society relies more and more on increasingly complex programs for moving people and information, and handling energy, defense, health and many other areas, it is paramount that we can rely on those programs, and our team has built a reputation for efficiently using formal reasoning to demonstrate their trustworthiness. Within the CEA LIST Institute, LSL is dedicated to inventing the best possible means to conduct formal verification. We design methods and tools (*Frama-C*[1], *Binsec*[2], *Unisim*[3], *etc.*), most of them open-source, that leverage state-of-the-art scientific expertise to ensure that real-world systems can comply with the highest safety and security standards. In doing so, we get to interact with the most creative people in academia and the industry.

Our organizational structure is simple: those who pioneer new concepts are the ones who get to implement them. We are a fourty-person team, and your work will have a direct and visible impact on the state of formal verification. Indeed, put together, the members of our team have a very mature experience with all of the most efficient formal methods (Abstract Interpretation, Weakest precondition calculus, satisfiability modulo theories, constraint programming, symbolic execution, *etc.*). The hired candidates will evolve in a very rich research environment, allowing them to master the necessary methods and apply them to their tasks. Furthermore, being at the frontier between academia and industry, the candidates can be sure that their work will actually be put to practical use with our partners. CEA LIST's offices are located at the heart of Campus Paris Saclay, in the largest European cluster of public and private research.

Workplace and benefits

Working at CEA will grant you an advantageous number of days off, holiday plans at a reduced fare and other material benefits through our CE ("Comite d'entreprise"). There is a company restaurant near the lab, accessible after a leisurely stroll near a forest populated by birds and squirrels. In summer, we also sometimes trade the company restaurant for a picnic in the nearby forest, where an area is dedicated to this, that includes natural climbing walls. There are also multiple fast foods and restaurants. The site is connected, either through the forest to the RER le Guichet, or through a series of highly reliable busses in dedicated lanes ("couloirs de bus"). There is also a service of electric bycicles covering the region.

[1] `http://frama-c.com`

[2] `https://binsec.github.io/`

[3] `http://unisim-vp.org/site/index.html`

In our modern building (less than 10 years), you will have an office with at most two other members. You are free to customize your working environment (almost) as you wish. We also have a package reception service, where you can receive your online purchases.

We are very keen on making the workplace as enjoyable as possible. We have regular coffee breaks during the day, fueled by free coffee and tea, providing a relaxed talking space between colleagues. Outside of office hours, we also organize numerous afterworks of all kinds: movie nights, board games and video games night sessions, group visits of various sites, *etc.* For the sports enthousiasts, most of the team members participate in one or more physical activities, ranging from football (usually on wednesdays at lunchtime), jogging in the forest, cycling, climbing (there is a club in Massy, near the RER station), and more.

In our team, the working atmosphere is treated with as much care as the work itself.

Professional perspectives at or after CEA

It is undisputable that AI is currently going through a rapid expansion. Several countries vowed to double or even triple (as is the case for France[4]) the number of students learning AI. This is not to say that the workfield would be saturated, because the demand for this expertise is fairly high. IT service companies would most likely always need manpower. However, the world of safety-critical systems (aviation, automotive, trains, energy, defense, health, *etc.*) is a considerable niche where experience with safety concerns and formal methods (in other words, employment in the LSL team) will offer a very competitive advantage. Even outside that niche, an experience in our lab can reassure future employers that you are fully aware of the good practices for the design of safe software.

Among other career options, several of our PhD students and other temporary collaborators have found a permanent position at CEA. Indeed, our hiring and growth has been well sustained for quite some time.

Problem

The field of Formal Methods (FM) may very well be one of the oldest fields in Computer Science, but it has been brought back to its infancy with the recent advances in Machine Learning (ML). The FM community spent decades perfecting the theories of its field, increasing its reach in industry, especially considering safety -critical systems such as transportation, energy and defense.

By pushing for higher and higher standards of safety, all the while providing the tools necessary to achieve these standards, it is *no understatement* to say that the FM community made the world a safer place [5]. Unfortunately, most of the FM lore falls short in tackling the cohort of new problems brought by the recent ML techniques [8]. Overcoming the opacity of the models, uncovering implicit properties and finding formalisms to specify them, detecting and repairing faulty behavior susceptible to cause significant harm, these are but a few of the large gaps in AI safety, for which there are no satisfactory solutions and for which FM are not adapted *yet*.

General description of our goal

Our lab seeks to adapt and expand its expertise to offer an adequate response to our industrial partners. Because this field is nascent, this demands both a fundamental research effort and its practical application in the design of industry-relevant tools.

This dual optic is the hallmark of our institution. Indeed, the hired candidates will be in a unique position: close to industry, to keep a real-world view of the field, as well as close to our academic partners and the research ecosystem of Paris-Saclay, allowing them to be grounded in robust scientific thinking.

Another desirable aspect is that the hired candidates will be at the forefront of a very new field, where the potential to leave a long-lasting impact is higher than in a mature field. Several directions of exploration are available, including:

- Specification: to formally verify a software, any software, one needs to first formally establish what this software is supposed to do. Indeed, we need to prove that the software satisfies a certain behaviour, and that behaviour needs to be carefuly described. Some of the properties can be defined for some Machine Learning artifacts. For example, in control & command systems, one

[4] https://www.lesechos.fr/2018/06/leffervescence-des-formations-en-intelligence-artificielle-997427

safety property could be “if an aircraft located at position P is approaching with speed S, and an angle A, the embedded neural network must emit a change in direction CD so as to avoid the approaching aircraft”. This type of property is easily expressible in mathematical notation because it relies on actual physical properties. However, a similar specification is not expressible for perception application. For example, the property “this neural network emits a brake directive when approaching a pedestrian” is not something that can be expressed because it is not clear how to specify what a “pedestrian” is. For traditional software, a considerable effort in ensuring safety is always dedicated to defining the formal properties of indispensable behavior (that must occur) and dangerous behavior (that must be avoided), in such a way that Formal Methods can prove them. We need a similar effort in Machine Learning. (see our paper [2]).

- Formal methods applied to test and verification: this is the heart of our efforts, around which the other research subtopics gravitate. Given the number of formal methods to apply, this direction in itself is approachable through many routes. We already have two on-going theses related to this, one is focused on exploratory research into formal verification of machine learning, the other partly aims at defining domains for our *Frama-C* tool that are especially powerful on the verification of neural networks. We also have several industrial partners supplying real-world use-cases. Finally, we also have a European project that partly aims at the verification of the low-level implementation of machine learning code.

- Scalability concerns: this is a more applied direction. There are already theoretically applicable methods with proofs-of-concepts, but they need significant improvement and novel and innovative adjustments to be able to become industry-relevant.

- Explanability: The opacity of Neural Networks is a considerable obstacle to their widespread adoption. We are also undertaking research in this direction through a thesis, in collaboration with Imag.

- AI for formal methods: orthogonal to the topic of “AI safety” is the topic of “AI *for* safety”. We also have several formal methods areas that can be improved by applying AI and are actively looking for candidates to help us in that direction. We currently have a European project, linked to our *Frama-C* tool partly dedicated to discovering new ways to tape into the potential of machine learning to improve formal verification of programs. We also have an on-going thesis, linked to our *Binsec* tool, on program constraints inference and verification.

Please note that the research topic of AI safety is vast and, while our team decided on a general roadmap to follow, we are *always* open to new ideas: if you believe you have a particular research subject you would like to pursue and that would be of interest to the general goal of AI safety, we will give it our full attention. If you have questions, don’t hesitate :)

Co-advisors and partners

An important tradition of our team is to encourage collaboration through co-advisement, so we invite researchers who are intrested in this topic and who can co-advise PhDs, PostDocs and internships to contact us to discuss it.

Related work

Efforts have been started internationaly in this topic. To give candidates an idea of what formal methods applied to AI would look like, we give references applying abstract interpretation to NN [7] and SVM [6], SMT to NN [4], symbolic execution to NN [3], and finally, this paper [1] which compares several approaches.

Application

While we are offering many positions (internships, PhDs, Postdocs, temporary and permanent positions), the workload and job description will evidently differ. For example, a permanent researcher would be evaluated on their ability to mentor students or to manage projects, while an intern would not. The demanded background is also evaluated differently. In any case, a non-negligeable part of the positions will involve coding. Varying abilities in the following areas (non exhaustively listed) would be the element of appreciation of your application:

- *OCaml* programming (at least, functional programming)
- *Python* programming
- AI in general and machine learning in particular. Specifically, through *TensorFlow* or *PyTorch*
- formal methods and static analysis (abstract interpretation, weakest precondition calculus, satisfiability modulo theoy, constraint programming, symbolic execution, *etc.*)
- strong mathematical background relevant to machine learning
- strong formal logic background relevant to formal methods
- mentoring
- software engineering
- formal specification
- project management
- academic English

Salary: academic competitive (vary *w.r.t.* diploma and former experience)

Availability: as soon as possible;

Contact: Please send an email to Isabelle Bontemps (Isabelle dot BONTEMPS (a) cea dot fr) and CC Zakaria Chihani (zakaria dot chihani (a) cea dot fr). Please indicate whether you are applying for an internship, a PhD, a postdoc or a researcher-engineer position.

Please join a detailed CV and, if possible, the grade sheets for your post-highschool years as well as reference letters. In your email or in a separate letter, please highlight in a few words how your background constitutes an advantage for the position.

References

[1] Rudy R Bunel, Ilker Turkaslan, Philip Torr, Pushmeet Kohli, and Pawan K Mudigonda. A unified view of piecewise linear neural network verification. In *Advances in Neural Information Processing Systems*, pages 4790–4799, 2018.

[2] Julien Girard-Satabin, Guillaume Charpiat, Zakaria Chihani, and Marc Schoenauer. Camus: A framework to build formal specifications for deep perception systems using simulators. *ECAI*, 2020.

[3] D. Gopinath, C. S. Pasareanu, K. Wang, M. Zhang, and S. Khurshid. Symbolic execution for attribution and attack synthesis in neural networks. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, pages 282–283, May 2019.

[4] Guy Katz, Clark Barrett, David Dill, Kyle Julian, and Mykel Kochenderfer. Reluplex: An efficient smt solver for verifying deep neural networks. *arXiv preprint arXiv:1702.01135*, 2017.

[5] Gerwin Klein, June Andronick, Matthew Fernandez, Ihor Kuz, Toby Murray, and Gernot Heiser. Formally verified software in the real world. *Communications of the ACM*, 61(10):68–77, 2018.

[6] Francesco Ranzato and Marco Zanella. Robustness verification of support vector machines. In *International Static Analysis Symposium*, pages 271–295. Springer, 2019.

[7] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. An abstract domain for certifying neural networks. *Proceedings of the ACM on Programming Languages*, 3(POPL):1–30, 2019.

[8] Emil Vassev. Safe artificial intelligence and formal methods. In *International Symposium on Leveraging Applications of Formal Methods*, pages 704–713. Springer, 2016.