A Learning-Based Testing Approach to Testing Software Applications under Evolution

In the context of an industrial research project, the Computer Science Laboratory in Grenoble (LIG) in France is opening a 3-year industrial PhD position (CIFRE contract) in the fields of Model-Driven Engineering (MDE), Model- and Learning-Based Testing (MBT/LBT), and Artificial Intelligence (AI). The project is about the refactoring, improvement, and testing of legacy software applications using AI.

1. Context

Developing complex and modular software systems following code-focused approaches is challenging. Model-Driven Engineering (MDE) techniques allow for raising the abstraction level from which developers build software. If the model is faulty, the errors will be propagated to the different refinements of the models downstream to the code. Therefore, finding and fixing bugs at model-level rather than in the code itself is crucial to develop robust applications.

Model-based testing (MTB) approaches [1] allows for generating test cases from a model of system requirements, but they require requirements to be available and executable. However, this is rarely the case when it comes to testing legacy software applications under refactoring and evolution. When such specifications exist, they are often expressed in natural language rather than in a formal languages exploitable by the machine.

Learning-based testing (LBT) approaches [2] allow for automatically testing black-box systems by feeding the system with various inputs and observing the outputs. Among others, model inference builds a model of a black-box system under test [3] based on a series of generated tests to understand the system itself. LBT is suitable in the context of non-regression testing of refactored applications as it permits to build two models of a software application, before and after refactoring, and to compare both. However, when it comes to testing new functions expressed in natural language, they are inefficient to generate test cases relevant for those new functions.

2. Thesis Description

The duties of the selected candidate will be to conduct research in the interdisciplinary fields of MDE, AI, and MBT/LBT. He or she will be in charge of: i) generating test suites to guarantee the absence of regressions in an application under refactoring ; ii) using MBT/LBT to generate relevant test cases to guarantee that new improvements in the refactored application satisfy AI-generated system requirements; and iii) writing up and publishing results in technical reports, or submissions to conferences and journals.

3. Qualification

The selected candidate must have a MSc degree in Computer Science or an equivalent degree with a strong focus on at least one of the three areas MDE, testing, and AI. We expect the student to be motivated, innovative, and interested in upskilling in the other areas.

4. Procedure

Please send the following documents to Damien.Pellier@imag.fr and nicolas.hili@univ-grenoble-alpes.fr:

- Your master's degree (M1 and M2) certificate in computer science/applied mathematics with your grades.

- Your CV.
- At least one letter of recommendation.
- Your master's thesis and any publications, if applicable.

Applications are processed continuously. You will be promptly notified by email regarding the status of your application and whether you are invited for an initial interview.

5. Links

- LIG laboratory: http://www.liglab.fr/
- MARVIN team: http://marvin.imag.fr/
- VASCO team: http://vasco.imag.fr/

6. Contact

- Damien Pellier <damien.pellier@univ-grenoble-alpes.fr> — UGA/MARVIN, Grenoble, France
- Nicolas Hili <nicolas.hili@univ-grenoble-alpes.fr> — UGA/VASCO, Grenoble, France
- Yves Ledru <yves.ledru@univ-grenoble-alpes.fr> — UGA/VASCO, Grenoble, France

7. References

[1] Mark Utting, Bruno Legeard. “Practical Model-Based Testing - A Tools Approach.” Morgan Kaufmann 2007, ISBN 978-0-12-372501-1

[2] Meinke, K., Niu, F., Sindhu, M. (2012). Learning-Based Software Testing: A Tutorial. In: Hähnle, R., Knoop, J., Margaria, T., Schreiner, D., Steffen, B. (eds) Leveraging Applications of Formal Methods, Verification, and Validation. ISoLA 2011. Communications in Computer and Information Science, vol 336. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34781-8_16

[3] M. Grand, H. Fiorino, D. Pellier: Retro-engineering state machines into PDDL domains. ICTAI 2020: 1186-1193.