EuroProofNet - Short-Term Scientific Mission

Title: Evaluation and development of the Alk platform for enhancing students' algorithm design and analysis skills

Applicant: Dr Oana Andrei, School of Computing Science, University of Glasgow, UK

Host: Prof Dorel Lucanu, Faculty of Informatics, "Al. I. Cuza" University, Iasi, Romania

Dates: 13 - 22 May 2024

Goals

With the rise of generative AI tools capable of developing algorithms for specific tasks, there is a heightened need to improve student skills in algorithm thinking, particularly in proving algorithms correctness. This proposal seeks to explore effective methodologies and teaching strategies to bolster these essential skills.

This collaboration has users of proof systems at the core. We claim that proof systems, along with the construction of proofs for algorithm properties through these systems, play a crucial role in fostering the development of algorithmic thinking skills. To illustrate this, we propose a detailed case study focusing on the specification and verification of algorithm properties within the Alk platform [1,2]. The ultimate objective of this research is to establish an integrated environment based on Alk, encapsulating both formal demonstrations of properties and incorporating AI tools.

The Alk platform serves as an educational tool aimed at facilitating algorithm design-and-analysis learning, fostering rigorous algorithmic thinking. Alk's symbolic engine [3,4] enables algorithm testing with symbolic values and property verification, such as invariants or contracts. Leveraging Matching Logic's theoretical foundation [5], Alk's interpreter, symbolic execution engine, and property checker are cohesively developed based on the language's formal definition.

Working plan

We will work together building on our previous collaborations on Rewriting Logic (the foundation of Matching Logic underlying Alk), similar expertise in formal fethods research and teaching algorithms and formal methods courses. An in-person visit will provide me the time to dive deep into understanding the foundations of and using Alk with the help of the Prof Lucanu who has been developing this platform and using it actively in the classroom.

Firstly, I will use my expertise in Computing Education Research the set up an evaluation study for Alk to investigate and explore how algorithmic thinking skills are supported and could be further expanded and strengthen when teaching an

algorithmics course. The main focus will be on skills relating to algorithm analysis, proof of correctness and complexity. We will work on refining the initial set of research questions to conduct a study on both students (at the host institution during the academic year 2024-25) and Computing Science instructors (from various Higher Education institutions worldwide). The ultimate goal is to gather evidence on the current capabilities of Alk, improve it further, thus contributing to the community of program correctness verification, both research and computer science education alike [WG3].

Secondly, we will work on investigating how the Alk platform can be used to develop algorithms in a rigorous way using Dijkstra's seminal ideas [6,7]. I will contribute to the existing Alk examples repository with gathering and designing new ones illustrating strategies for developing algorithms using Alk, as well as revising the existing ones. The ultimate goal is to develop further the existing Alk Primer [8] by showcasing the automatic property checking capability and making the library of examples readily available for the wider CS community [WG4].

Thirdly, we will work on how Generative AI tool (such as ChatGPT) may be used to provide algorithmic solutions to given problems and then use Alk to validate and prove the correctness of the proposed solution or fix it if not correct [WG5]. We aim to address the question of how to synthesise formal specifications from the explanations given by the Generative AI tools to prove their correctness [WG3]. Can this be done automatically? The ultimate goal is to integrate generative AI functionalities in Alk. Currently I am conducting research on using Generative AI as the basis of a collaborative learning tool for tracing algorithms, now in the final phase of analysing data gathered from a recent usability study.

Expected outputs and contribution

We present the expected outputs in relation to research-coordination objectives (RCOs) and capacity-building objectives (CBOs) of the Action MoU:

- (RCO3) "Make techniques for program verification more effective and more accessible to all stakeholders."
  We will be developing the Alk platform further, currently listed as a tool in the EuroProofNet Program Verification inventory. [WG3, D8]
  https://github.com/EuroProofNet/ProgramVerification/wiki/List-of-tools

- *(RCO6) "Develop the use of artificial intelligence and machine learning techniques on proofs."*
  We will work on synthesising formal specification from explanations provided by Generative AI tools in Alk and proving their correctness. [WG5, D12]

- *(CBO4) "Ease access to formal verification techniques in education and other areas of science."*
  We will prepare detailed documentation for the Alk platform, building upon the existing primer and set of examples. We will prepare to run a study on evaluating

the platform's capabilities in supporting the enhancement of students' algorithm design and analysis skills; the results will be submitted to Computing Education conference (ACM SIGCSE events) and/or formal methods conferences. The current github repository [9] is maintained and will be updated accordingly. [WG4, D11]

- *(CBO5) "Actively support young researchers, the under-represented gender, and teams from regions with less capacity."*
  While I have been a researcher on formal methods for almost 20 years, in 2020 I have started a lecturer (assistant professor) position in the School of Computing Science, University of Glasgow, focussed on Learning, Teaching and Scholarship. Since 2021 I have been working in Computer Education research with particular interest in teaching algorithms and formal methods (UKICER'22, ICER'23) and competency-based education (SIGCSE TS'24). Working with Prof Lucanu on algorithmic thinking and the Alk platform will allow me to develop further my expertise and scholarship in and contribute to both Computing Education research and the formal proofs research area.

- *(CBO6) "Transfer knowledge in terms of expertise, scientific tools and human resources across the different disciplines and between academia and industry."*
  A particular contribution to this collaboration is my expertise in computing education research and higher education (currently enrolled in a Master of Education program).
  As Deputy Director of the Graduate Apprenticeship in Software Engineering program in the School of Computing Science and teaching the Algorithmics course on this program, I can transfer knowledge related to designing and proving algorithm properties in Alk to current students and graduate, as well as gather feedback from employers in the Software Engineering industry associated with this program.

- *(CBO7) "Prepare competitive EU researchers for a fruitful career in an international environment through intensive use of STSM and joint educational programs with industry."*
  The funding landscape available for my type of academic role (teaching-focussed) is extremely limited, hence an STSM will provide me with the opportunity to conduct research that will enhance my career trajectory as well as to propose and supervise student projects on the topics covered by this collaboration.

- *(CBO8) "Disseminate the results of the Action activities to the scientific community, the industry, the certification bodies, the European institutions and to the general public."*
  We aim to publish the results of this work in ACM SIGCSE conferences (e.g., TS, ICER, ITiCSE, UKICER, CEP), theoretical computer science conferences and journals (e.g., ICTAC, FM, TASE, JLAMP).

[1] Alexandru-Ioan Lungu and Dorel Lucanu. Alk Language reference manual. https://github.com/alk-language/java-semantics/wiki/. Accessed: 1 March 2024.

[2] Alexandru-Ioan Lungu and Dorel Lucanu. Alk platform. https://github.com/alk-language/java-semantics. Accessed: 1 March 2024.

[3] Alexandru-Ioan Lungu and Dorel Lucanu. A matching logic foundation for Alk. In Helmut Seidl, Zhiming Liu, and Corina S. Pasareanu, editors, Theoretical Aspects of Computing - ICTAC 2022 - 19th International Colloquium, Tbilisi, Georgia, September 27-29, 2022, Proceedings, volume 13572 of Lecture Notes in Computer Science, pages 290–304. Springer, 2022.

[4] Alexandru-Ioan Lungu and Dorel Lucanu. Supporting algorithm analysis with symbolic execution in Alk. In Yamine Aït Ameur and Florin Craciun, editors, Theoretical Aspects of Software Engineering - 16th International Symposium, TASE 2022, Proceedings, volume 13299 of Lecture Notes in Computer Science, pages 406–423. Springer, 2022.

[5] Xiaohong Chen, Dorel Lucanu, and Grigore Rosu. Matching logic explained. Journal of Logical and Algebraic Methods in Programming, 120:100638, 2021.

[6] Edsger W. Dijkstra. Guarded commands, non-determinacy and formal derivation of programs. Comm. ACM, 18(8):453–457, 1975.

[7] Edsger W. Dijkstra. On the teaching of programming, i.e. on the teaching of thinking. (circulated privately). Feb. 1975. http://www.cs.utexas.edu/users/EWD/ewd04xx/EWD473.PDF.

[8] Dorel Lucanu. Alk Primer – Java-Semantics Edition. 2020. https://github.com/alk-language/java-semantics/blob/master/doc/alk.pdf

[9] Alk GitHub repository. https://github.com/alk-language/java-semantics