Boosting Automated Verification Using Cyclic Proof: Pathways to Impact

J. Brotherston (PI), B. Cook (CI), N. Gorogiannis (RCI)

This proposal is about using cyclic proof technology to develop advanced methods for solving a range of problems involving inductively defined predicates — such as entailment, frame inference, termination or safety analysis, and definition abduction — and integrating these within the framework of a large-scale program analysis such as SpaceInvader or JStar. We believe that by doing so, we are likely to achieve a significant boost in the automation and coverage of such large-scale analyses. Such an improvement would have a real and substantial impact on the boundaries of the state of the art in automatic program verification, and might eventually lead to our techniques being taken up in industrial tools such as those being developed by Microsoft or Monoidics, our two Project Partners.

Based upon the above, we believe that the primary means of maximising the social and economic benefits of our research will be to ensure that our ideas are exposed as widely as possible to, and hopefully taken up by, academics and industrial researchers working on the automatic verification of real-life code. We have identified a number of pathways that we will utilise during the project in order to ensure that our ideas and results are disseminated to such researchers:

Pathway 1: Academic publication

As a matter of course, during the project, we will aim to regularly produce papers describing our research, and publish them in reputable venues. In computer science, one normally targets conferences rather than journals as the primary publication venues in the first instance: Conference publication is normally much more timely than journal publication, and is seen as having much higher impact (at least initially) since it entails direct engagement with the conference community through a public presentation of the paper and networking with colleagues at the conference. For similar reasons, it might also be beneficial for a team member to attend a high-profile verification conference even in the case that they are not presenting a full paper; many conferences run poster sessions or short paper sessions precisely for the showcasing of work that is still in progress. We will try to ensure that the conference travel budget is utilised as fully as possible to maximise the exposure of our work to the appropriate academic communities.

Pathway 2: Academic collaboration

There are several research groups in the UK whose research interests encompass program verification; we mention as prominent examples the Automated Reasoning Group at Cambridge, the LFCS in Edinburgh and the Verification Group at Oxford. We will offer to visit these groups in order to give research seminars on our work, and also invite researchers from these groups to come and give seminars on their own work at UCL, where the PPLV group runs its own regular seminar series. We hope that this will facilitate knowledge exchange, lead to cross-fertilisation between research areas, and lead to collaborations between these other groups and our own.

Of course, there are also a number of research groups in Europe, the US and Asia who also work on program verification, including groups in CMU, Singapore, Copenhagen, and INRIA (France), amongst others. We will of course attempt to undertake collaborative activities with these groups in much the same manner as described above for the UK groups. However, the considerably higher financial and time costs attached to European or worldwide visits are likely to mean that the greater part of such communication will be electronic rather than face-to-face (which is another reason why conference attendance is so useful).

Pathway 3: Prototype tool development & dissemination

In this proposal we place a significant emphasis on the development of software implementations of our techniques, which are vital as a proof-of-concept and in investigating how these techniques can be made to work efficiently within the context of an interprocedural program analysis running on medium-to-large sized code bases. We will also maintain and seek to continually expand test suites that benchmark the performance of our tools in isolation and

in this wider verification context. Both the tools and the test suites will be made publicly available on the WWW, as well as being advertised in our research papers.

Pathway 4: Industrial collaboration

Ultimately, we believe that the litmus test of our cyclic proof methods will be the extent to which they can be used to boost large-scale program verification. We have identified two Project Partners — namely, Monoidics Inc. and Microsoft Research Cambridge — who are actively involved in the research and development of such program verification tools[1]. Both partners stand to benefit from the boosts in automation and program coverage that we hope will result from the research outlined in this proposal, and have volunteered to provide time in discussion and, hopefully, collaboration with the project team. We expect to meet with both partners at least twice per year during the project lifespan in order to discuss the direction and potential applications of our cyclic verification techniques, and how they might be integrated into the industrial program analyses being actively developed by both businesses. We note that the partnerships are particularly convenient due to the close geographical proximity of both MSR and Monoidics to UCL; we hope that this proximity will help to foster back-and-forth communication between the project team and our two industrial partners.

Resources

No extra resources are requested for the activities outlined above. The costs of attendance at conferences and research visits to Project Partners and other collaborators will be met from the overall travel and subsistence budget (see Justification of Resources). We expect that Project Partners will provide funding for their researchers to visit the PPLV group where appropriate.

[1] We believe it is of definite value to have obtained the support of several expert researchers from Microsoft Research, even though Prof. Cook is cross-appointed there. He is named as the Co-Investigator on this proposal, and will receive support, in his capacity as a UCL professor rather than as an MSR employee.