Homotopy Programming and Verification — Pathways to Impact

Cost of Software Failure: The cost of software failure is truly staggering. Well known individual cases include the Mars Climate Orbiter failure (£80 million), Ariane Rocket disaster (£350 million), Pentium Chip Division failure (£300 million), and more recently the heartbleed bug (upto £250K per server). There are many, many more examples. Even worse, other software failures such as one in the Patriot Missile System and another in the Therac-25 radiation system have costs lives. More generally, a 2008 study by the US government estimated that faulty software costs the US economy £40 billion annually. As a result, the human and economic importance of ensuring programs run without error is hard to over-estimate. Further, this importance will only grow significantly as software becomes ever more ubiquitous in our lives and economy.

Formal verification uses mathematical techniques to prove that programs actually perform the computations they are intended to perform and/or avoid performing unintended ones. One major approach to software correctness is *language-based verification*, in which successful compilation of programs provides machine certification of their correctness. Language-based verification thus supports the development of software that is *correct by construction*, which is the logical conclusion of the persistent trend in software engineering toward ever earlier program verification; examples of their use are given in the main body of the proposal. Although the proposed research — namely, the construction of the first of a new breed of programming language and verification environments based upon HoTT — does not lend itself to immediate industrial uptake, we have devised the following five-point plan for maximising its impact:

• **Publication:** We will, of course, pursue the kinds of publication venues all good scientists pursue. Principally, we will produce high-quality papers and endeavour to publish them in the best journals. Publication of our papers in leading archival journals will confer validation by the community of the correctness and importance our our results, as well as allow them to serve as seminal references. However, the tradition in computer science is also to aim for early publication of important results to keep pace with the rapidly changing nature of the discipline. We will therefore also seek to publish our key results in top conferences. We expect each of our work packages to result in at least two publications and several to result in more.

• **Scientific Interaction:** To increase the impact of our research, we will continue to interact with our research community via conferences, research visits, and other meetings. For example, within the UK, Prof Ghani co-founded the Scottish Category Theory Seminar while Dr McBride co-founded Fun in the Afternoon. They are both active members of the Scottish Programming Languages Seminar and SICSA, the Scottish computer science pooling body. More generally, Scotland is an excellent place to interact with other researchers on a regular informal basis: there are strong programming languages and verification research groups at Edinburgh, Glasgow, Heriot Watt, and St Andrews, and there are also Scottish Theorem Proving meetings. Further afield, we will interact with researchers from across the UK. We have excellent contacts at all the major relevant research groups, and although distance will make interactions with them less frequent, that in itself will provide fresh perspectives as the grant progresses.

At a more detailed level, we are already committed to host a specialist workshop in HoTT in Strathclyde towards the end of 2014 which has been partially funded by the LMS. This workshop is exactly the kind of high-impact event — targeted at a small number of experts and in a very specific area — that truly aids and disseminates research. Our experience is that by careful choice of time and location, we can attract significant numbers from both Europe and further afield to such workshops and the attractiveness of our workshops will be enhanced by inviting key leaders in the field. Because of the value of such workshops, we intend to hold a follow-up workshop during the project and have therefore requested funding for this within the proposal budget. The cost is entirely minimal compared to the opportunities such focussed meetings offer.

• **Collaboration:** To further maximise impact, we have invited a number of internationally leading project partners to work with us on specific work packages of the proposed research (see below). This is advantageous in several ways. First, working with these project partners will enhance the scientific quality of the proposed research. Secondly, working with these partners will mean that they are intimately involved with the proposed research; in effect, thorough dissemination of our ideas will be occurring with our partners even as the work is being done, and this will help get our ideas out into the broader research community. Finally, drawing on a variety of different perspectives will help ensure that the proposed research does not become overly insular, but instead is outward-looking and solves important problems in key application areas.

Our project partners are Prof Steve Awodey(CMU, WP1), Prof Thierry Coquand (Chalmers, WP2), ?? (??, WP3), Dr Nick Benton (Microsoft, WP4), ?? (??, WP5), ?? (??, WP6), ?? (??, WP7) and Dr Andrew Kennedy (Microsoft, WP8). We have thus secured the involvement of an industrial partner for each of the impact generating work packages in. In terms of maximising impact, the collaboration with Kennedy is the closest we come to actual industrial crossover. Since Kennedy's original units of measure system has been incorporated into Microsoft's *F#*, the generalisation we propose to develop may do as well. Although it is not the main focus of the proposed research, we will certainly investigate this possibility.

• **The Stream Model:** One prevalent model of research is the *stream model.* Under this model, to solve a problem one looks both "upstream" for fundamental theoretical ideas to feed into the research, and "downstream" for validation of the research by those who stand to benefit as end-users. This model maximises impact by ensuring that research is picked up not only by those working on the same problem, but also by those upstream, who will be interested in how their own work is applied, and by those downstream, who will look to apply it to their work. To help ensure that the proposed research has a high impact we will follow this stream model.

Upstream of us there is a significant number of mathematicians and theoretical computer scientists working on the foundations of HoTT, and whose results we will enhance and apply to develop our environment for programming with HoTT. Downstream, those in the programming languages and program verification communities are already interested in applying HoTT to their work, but the lack of an actual programming language hinders them from doing so. Thus our tools will be exactly what they need to progress the uptake of HoTT by the broader programming languages community.

• **The Work Packages:** Our main consideration in developing our work packages is the quality of the scientific ideas underlying them, since high-impact research obviously requires high-quality ideas. For the specific research proposed here it also requires i) not just a much better understanding of the current literature on HoTT but substantial contributions to it; ii) not just theoretical ideas presented in a way that theoreticians can understand, but also concrete program languages and verification tools that programmers can understand and use on their own terms; and iii) not just evidence of the overall power of our new tools, but also a collection of examples outlining a methodology for deploying them. These three requirements guided our designing the work packages. WP1-3 make significant contributions to the literature on HoTT while WP4 generates impact from these work packages by developing examples to illustrate their results. Similarly, WP5-7 focusses taking the theoretical results we will achieve and using them to build our tools. Finally WP8 generates impact from these tools by developing case studies where they are used thereby making them directly usable by non-specialists.

• **Hidden Foundations:** Although the language and verification tools produced will be based on HoTT, what users of our tools will see is a clean, modern programming language whose use requires no knowledge of the underlying foundations of HoTT. We have already applied this "hidden foundations" approach to mathematical descriptions of programming language artefacts where the mathematics is used to organise the structure of a programming language but not used in programming itself. However, even in the foundational work packages, we will still use the hidden foundations approach, e.g., by comprehensively treating special models of interest and by showcasing our results via examples as much as possible. This will allow those with modest or no mathematical background to use and profit from our all of our research to the best extent possible.

References

[1] Cost of Independent Software Verification & Validation. 2011. At `http://www.galorath.com/wp/cost-of-independent-software-verification-validation-ivv.php`

[2] Total economic cost of insecure software: $180 billion a year in the U.S. 2008. At `http://news.cnet.com/8301-13846_3-9978812-62.html`