Planning a (CS) Research Project for PhD studies by Prof. Abhik Roychoudhury, National University of Singapore, March 2018

Discussions undertaken in CS6210 class on 8th March 2018

Suppose you are planning a research project as part of your PhD studies. What are the sanity checks you might undergo? There are lot of opinions in this topic, and there is no one answer. In the following, certain scenarios are painted, which may be useful to reflect upon. In the end, the situation of each research project may be different. Even before we start giving out the points, it is useful to mention the issue of fit. You could do an exercise of why you or your research group is the right person to solve the problem at hand, what unique perspectives you have, and what others do not have. If this exercise goes well, we can consider the following points.

First you could think of motivation of the research area, who would be interested in it, and who would fund such research. This is a basic sanity check. Note that this check does not preclude theoretical research by any means. Secondly, you could also think about the future outlook of your research are, how it flows with the technology growth in CS that you see. Related to this, you may also think, just as a thought experiment, what is the timeline of deployment of your research, if you were to spend a lifetime on it. It can be a short time like 5 years, or it can 50 years, but it is still useful to go through this mental process. It gives clarity of purpose in terms of the kind of techniques you may include in the solution you propose.

Next, you have to capture and suitably answer whether your proposed work (and nugget of a solution you have in mind) will satisfy the intellectual curiosity of others, e.g. that of an intelligent outsider. This may appear to be unimportant for doing research, but it is not. Even to have a high impact publication in a major flagship CS conference, this may be a useful exercise to go through. My own research is in Software Engineering, but my expertise is in program analysis for example. So, if my research is to be of interest to people in software engineering, but outside program analysis, I should be able to appeal to their imagination. This exercise is useful anyway, since the intelligence (solution capability) and imagination (freshness of ideas) are both two important aspects of research.

Next, you might want to focus on the contributions of the work that you seek to bring to the world, the *conceptual* contribution, *technical* contribution and *utilitarian* contribution. These we have discussed earlier. For conceptual contribution, you may want to do the exercise on novelty of the planned work thoroughly, and the fresh outlook you are bringing. For technical contribution, you may want to see whether you have the necessary technical training from your research group or your undergraduate studies, and if necessary take appropriate modules. For utilitarian contribution, it refers to the implementation set up and experiments you will work on. You need to be careful in this. In certain research areas of CS (including my own), it is difficult to build a tool from scratch, and you can achieve your goal faster by sourcing open source tools or homegrown tools previously built in your research group. It should be noted that experiments and implementation are often the focus and worry of many of our PhD students – but it is only one of the many factors that contribute to success of your planned research project.

Timing is important

When you bring or propose an idea into a research community is important. In other words, it is not enough to propose a good idea, but the ground should also be fertile and ready for your idea. If the research community is several steps behind you, the idea may be dismissed as too futuristic.

Do note that the research community can also be several steps ahead of you, though you have a good idea. How is this possible? It is often the case, that the broad idea might have existed in certain sub-communities, but no real solution or technology exists. So, even when you bring in a new technology, this may be dismissed as implementation of a known idea. There could be a thin line between a breakthrough practical solution, or a solution which implements a known idea. This is a thin line, and how the research community views your work, will depend on the timing, when you decide to come forward with your idea and solution.

It may work in your favor, if the research community is few steps behind your proposed work, but not too many steps. Then you propose an innovative solution with a new technology with some justification of why this innovative solution is feasible now, and it was not feasible earlier. Just as an example, since Satisfiability modulo theory (SMT) solvers have become mature, one may propose program analysis solutions based on SMT constraint solving, and argue that such solutions were not feasible earlier.

Perception is important

You work very hard on a paper but then it is turned down, or it is accepted but does not get great reception. May be you give a talk in a conference and the audience reception is not that great, and you feel disheartened. There is as such no reason to feel disheartened. There could be a gap between your perception of the work and how the community views it. This gap cannot always be bridged but it is useful to know and discuss this.

One of the problems in perception is often the perceived lack of fit. While this can happen for many reasons, one common reason is "fitting a solution to a problem" syndrome. Suppose I am working on software testing using symbolic analysis. Here software testing is the problem, and symbolic analysis is the solution technology. I could have expertise in symbolic analysis, but yet my contribution may be unwelcome, if I do not demonstrate knowledge about nuances about the problem I am seeking to solve, namely software testing.

Sometimes, the community may feel that a proposed research is incremental, but as a researcher you do not feel that way. Why could this happen? This could happen due to "this is easily achievable" syndrome, because an idea has existed but it has not been brought forward by the research community in the form of a solution, and your work is bringing it forward. In such cases, sometimes an experimental head-to-head comparison with existing state-of-the-art technology may help, but the nuances of how it will be perceived by the research community is important too.

Finally, researchers may have implicit preferences about certain technologies based on past experiences. This is the "been there done that" syndrome. So, they could have tried a technique (say model checking) in the past for their problems and may be it did not work for scalability reasons. Meanwhile model checking technology has improved, and you are using the improved technology in your research, but somebody may comment that your solution will not scale. It is useful to be aware of such nuances and dispel such potentially pre-conceived notions in a concrete fashion.