A Peek To The Ph.D. Experience

— After reading Philip Guo's "The Ph.D. Grind"

5 February 2018, Ziqi Zhang

It is very educational to read about another Ph.D.'s experience and learn some lessons from it, especially an experience from such an outstanding graduate in Stanford as Philip Guo. In his book "The Ph.D. Grind", he recalled his grinding Ph.D. life and, what's important to me, summarized several lessons for new Ph.D candidates. After reading this book, I not only learned several correct attitudes I should take towards my future research life, but also many precautions I should care in my future life. What's more, there are many small points that I appreciate a lot.

In order to take these valuable things down for the future, I write this article to record the lessons I learned from this autobiography. Another purpose of writing this article is to share my understandings with my friends and I'm very happy and open to talk with anyone who has also read that book.

I will firstly sort Philip's research experience during PhD life. Then I'm going to write some lessons about the right attitude I should take in Phd life and some bad situations I should pay attention to. In the end, I will write some other lessons I learned to mark small points worthy remember.

A Brief Summary of Philip's Research Experience

Firstly, to be methodical, I'd like to sort out Philip's research experience during his Ph.D. life because I think following the time line is the most obvious and efficacious way to recall what I thought when reading his book. By the time order, Philip Guo's research projects include following:

- First four months of PhD — Philip used Klee to find bug in the codes of Linux device drives, which was a subproject attached to a senior PhD's paper. The group working on that paper was large but the acceptance result was not good because they didn't have time to polish their paper. Philip suffered a lot during that experience.
- Ten weeks from Apr. 2007 — Philip tried to get out of the previous project and spent a lot of time lonely meandering, which later he found to be a bad choice.
- Summer of 2007 — Philip was an intern at Google.

• First half of the second year — Philip collaborated with Scott, a young HCI professor and Joel, Scott's graduate, for a top-tier HCI conference and it was accepted. That was a successful experience because both two of his collaborators were insiders of that subfield.

• Another half of the second year — Philip worked on empirical software measurement with his advisor Dawson. Nonetheless, Philip and Dawson had plenty of trouble publishing their results because they were not insiders in that subfield.

• Beginning of the third year, July to October, 2008 — Philip worked about a cross-checking project based on Klee. They never submitted the paper because his lack of technical expertise and insufficient mentorship.

• Middle of third year, to Jan. 2009 — Philip tried to improve Klee search algorithm. After months of working, he found out that a tiny improvement of accuracy, which would only come true after his months of hardworking, was not novel enough for a good paper and he couldn't find the breakthrough.

• Feb to Mar, 2009 — Philip worked on Klee UC with a new PhD, Peter. They didn't figure it out and quitted and that was the last time he had any relation with Klee.

• Summer of 2009 — Philip was an intern at MSR. He work on some people-related factors that effect the final results of bug reports. It was also empirical software measurement but what's contrary to his former cooperation with Dawson, his paper was accepted by a top-tier conference due to MSR's enormous data base and Tom's brilliant writing skills.

• Fourth year — Philip worked on the IncPy project, which was his first idea based on his former internship experience and finally resulted in a workshop paper and a conference paper. He firstly implemented it, then submitted a workshop paper, then tried hardly to find users for his experiment, which was not successful, and finally, submitted a paper which was rejected.

• Beginning of fifth year, Jul to Sep, 2010 — Philip worked on SlopPy, which was also his own idea and turned out into a workshop paper.

• Oct. 2010 — Philip worked CDE, which was his own idea and turned to be a very successful tool. CDE was used by thousands of researchers after his polish for a very long time.

• Summer of 2011 — Philip was an intern at Google working at CDE.

• End of fifth year — Philip worked on ProWragler, a project that he collaborated with a young professor, Jeff and it successfully turned into a top-tier paper.

• First four month of sixth year — Philip worked on Burrito with a MIT professor, Margo and that was a comparatively successful project experience but the paper was rejected by a top-tier conference and turned to be a workshop paper.

Attitude Lessons

After so many ups and downs in Philip's research experience, he finally found his own way and made it to graduated. As a new PhD student, there are so many lessons to learn from his experience and I'd like to mark down some right attitude I should learn from Philip when facing difficult situations.

The first attitude I should learn from Philip is to be a good solider and pay the dues, which means I should obey my advisor's will to do some projects as a junior lab member. Philip thinks it is an unavoidable phrase to enter a lab for that, as far as I'm concerned, it can make my advisor happy and have his endorsement. It's impossible for my advisor to back me when I don't want to do anything pleasing him or her. Sounds ugly but it is reasonable.

The second attitude I learned from this book is always to be proactive to speak to professors. Don't feel shy or embarrassed. As for this aspect, Philip did something right but also did something wrong. There are two aspects that I should really ask advice from professors about, one is about my current experience and my research problem, and the other is about collaboration. Previously one of my biggest problems was not dare to chat with professors. But from this book I learned that there is nothing to worry about chatting with them. Even if I didn't perform well in a project with the professor, professors are also open to talk with me. Also, even though I don't find any opportunity to collaborate, there is nothing to worry about. I don't lose any thing. Further more, it is possible that he will recommend me to a professor who share the same research interest with me, which is a great chance for me.

The third attitude is to accept failure calmly. Philip met tons of failures, including that he didn't finish a project, he failed to find the cooperation point with a professor in a talk and he disappointed some potential users of his software. When facing these failure, he didn't complain or disappoint, instead he calmly accepted it and tried to continue his research. Don't focus the attention on past failure but on future plan. That's a very valuable lesson I learned from him.

Bad Situations to Avoid

Besides some precious attitudes lessons, another important thing I learned from this book is some traps, or some bad situations about collaborators I should avoid.

The first dangerous thing is to try a subfield when neither of me or my colleagues are insiders in that subfield. In this book, there is a sharp comparison between Philip's collaboration with Dawson, not an insider and Tom, an insider in the same subfield. The project with Tom was very successful, contrary to that with Dawson. When submitting a paper to a conference, insiders know the rhetorics, innovative points and many other writing skills that appeal to reviewers. With these refinement, the possibility of a paper to be accepted increases a lot. I don't mean that outsiders have no way to publish a paper. What I emphasize is that with an insider's help, the road to a publication can be smoother and faster, which means I don't have to pay for excess grind.

The second important thing is that the collaborator should pay attention to this work. It is dangerous if my collaborators are not enthusiastic enough, not interested in the project or don't regard this project important to his current career stage. It will be effective for me if I can find a young professor who is eager to publish papers as a collaborator. Philip's cross-checking project is a perfect example. His advisor was not very enthusiastic about this idea and his senior student collaborator was busy finding a faculty job, which meant one more paper was not significant to his current career stage. As a result, he quitted that project partly due to the lack of mentorship.

Other Lessons

A Truth about Research

There is a cruel truth about the difference between the success of a project and the success of an individual graduate student. It may take several failure attempts of former graduates to finally make a successful project. A professor might need to go through several rounds of student failures and dropouts before one set of students eventually succeed. Sometimes the lifetime of a project lasts longer than a PhD's lifetime. From the professors' point of view, as long as the original vision is realized and published, then the project is considered a success. However, for the intermediate PhD who pays thousands of hours and suffers a lot psychologically only to quit that program, it's hardly to say this is a successful project.

Professor's Interest Matters

It's important that my work aligns to the advisor's interest for that if not, it's difficult for the professor to enthusiastically and frequently give advice. I should more or less appeal to my advisor's interest. So to find professor's interest, the most effective way is to read his or her recent papers and grand applications, just as Philip did when he was going to work with Margo. He read Margo's recent papers and grand applications to learn about Margo's research interests and finally found a good project that fitted both of them.

How an Original Idea Becomes a Publication

In the part of his sixth year, Philip recorded carefully how the simple idea became a publication. This process includes sketching high-level plans, figuring out preliminary detail plans and finally, low-level hardworking. It's important to sketch out the outline of the project and details before submerged by tons of work.