Survey of the Nature and Tools of Computer Science Research

Amanda Wulf
The University of Texas at San Antonio
1 UTSA Circle
San Antonio, TX 78249
amandakwulf@gmail.com

ABSTRACT
A survey of the nature and tools of computer science research. Topics discussed include how computer science research is similar to and differs from other fields, and the various things a good computer science researcher needs to know.

Categories and Subject Descriptors
10002944.10011122.10002945 [**General and reference**]: Surveys and overviews

1. INTRODUCTION
Computer science is a young field that has some similarities to and differences from other research fields. People interested in computer science research need to learn these, as well as how to go about research, what tools they will need, and qualities they should develop in themselves. This paper seeks to address these issues.

2. THE NATURE OF RESEARCH
2.1 Research in General
Research is the process of discovering or creating new ideas in a certain field. It's undefined and open-ended groping in the dark [10]. Most of the time, each research project or paper is a small, incremental step forward in the field [1]. Research is never "complete" in one field [1], it is always continuing. It focuses primarily on formulating things [15].

It's important because researchers' promotion and grant money is tied to the number of publications they make [6], so if you want to be a computer scientist you should be prepared to do lots of it.

There are two types of research: Finding a problem, and finding a solution [8].

2.2 Computer Science Research
The categories of computer science research include the following:

- Problem-solving concepts
- Computer concepts
- Systems/software concepts
- Data/information concepts
- Problem-domain-specific concepts
- Systems/software management concepts
- Organizational concepts
- Societal concepts
- Disciplinary issues

[15]

There are several differences between computer science research and general scientific research. Computer science research largely focuses on technical analysis, as opposed to society or profession topics. It's much more focused on formulating processes, methods, and algorithms than anything else [15]. It proposes many new entities, but it's generally not concerned with doing much evaluation or validation of new designs [5].

Computer Science research does not really involve the standard scientific method. Instead, it's based more on intuition developed from lots of paper reading, prototyping, and experience [4].

3. THE TOOLS NEEDED IN CS RESEARCH
3.1 Research Process
3.1.1 Beginning Your Research
The first thing you should do when beginning research is to pick a topic. The only way you can pick a research topic is to know your research interest and sub-interest [8]. If you don't know your interests yet, the only way to find out is to do lots of reading in general in your chosen field. You can also look at hot areas and learn about open problems [7]. Make sure it's a real-world need [11]. Think of a question you want to answer, then narrow it down into a small, solvable problem [2]. Try to find an unsolved problem in your lit review [10].

Once you find your research topic, the first thing you should do is study the background of your chosen topic by studying any textbooks related to background knowledge needed as well [9] [11]. Once you have that knowledge, the majority of research experts recommend that the first thing you should do is a literary review of any and all papers related to your topic that you can find. You should also talk to people who have worked or are working on something similar [4]. The reason you should do this is because it avoids the dead ends inherent in pursuing research that someone else has proven doesn't work [8] and reinventing the wheel by re-discovering an idea someone else has already published [8] [4].

Next, you should begin prototyping [12]. Analyze your prototype by collecting and generating data and analyzing this data using statistical tools [12] [8].

Finally, you must publish your work. You should select a venue where related work is often published or top conferences in your area in order to enhance your paper's chances of getting accepted [16].

3.1.2 General Advice

For some projects, you should consider collaborating with other research. Computer science research often involves collaboration due to the complex nature of the work, with large prototypes often needing to be produced. Computer science averages three coauthors per paper [6]. You must collaborate on large projects if you ever hope to finish them [8]. Grants often favor collaboration as well [8].

Researchers also often have find funding. They must write grant papers explaining why their work is important, presenting their ideas clearly, and positioning their ideas against other work [16].

According to Robbins, one of the biggest challenges in research is having enough time to get it all done. You can be efficient by having dedicated distraction-free research/writing time daily [12]. Never "put your research aside" for awhile, always work on it every day. Be prepared to write down ideas that might pop into your head at any moment [14]. When you're coming up on a deadline, you might have to spend more time working for awhile, but in general try to plan to be finished with your research paper at least two to three weeks before the deadline [14].

When researching, don't give up hope if you run into obstacles along the way. If you miss a deadline, submit your paper to the next conference [14]. If you're facing a big problem you can't solve, keep at it. After a lot of effort you might get there [10]. Finally, if you hit a total dead end in your research, don't be discouraged. You can use those findings from the dead-end research project in other research [8].

3.2 Literary Reviews

Before picking your research topic, while picking your research topic, and while doing your research you should be doing a literary review. Read as much as you can on your proposed or current topic [3]. Read consistently for a certain amount of time each day. You might not get as much reading done as you'd like at first, but you'll slowly build up stamina over time [8].

If you're not sure where to start, read top conference papers from IEEE and ACM [8]. Also, make sure you're reading papers with high citations or ones written by well-known authors [12]. Use scholarly databases over Google to search for papers, as higher quality papers can be found this way. Use reference management software [3].

While reading papers, you should first work to understand the paper, then think about it critically [8]. In your reading, differentiate between partially and totally solved problems [10] and argumentative vs. analytical papers [3]. Make sure to take notes on references and keywords [3].

3.3 Advice for Novice Researchers

Novice reserchers should firstly understand their own field. They can start by reading (a lot) most or all of the classic textbook in their subfield [2] and reading papers in their subfield (both recents and classics, especially ones with many citations [8]). They should also read and follow the research of leaders in their subfield [11], and the "hot" areas of their subfields [13]. Reading even one paper might take multiple days at first, but they should slowly build up to the point of reading one to two papers per day.

They can develop their reading comprehension skills by keeping a research journal [11] in which they summarize book chapters and papers in their own words, implementing things discussed, and testing their implementations [12]. If they're in a subfield that involves proofs they should also practice reading and writing proofs for things they're reading about [16]. They can also go to their advisors for help if they get stuck when trying to understand a paper [4].

Once they understand their own subfield, they should work on getting a good understanding of subfields related to their subfield [8].

They should also be networking by going to conferences and talking to people working in their area [8]. They can work on their presentation skills by presenting their current research at conferences.

They should be prototyping and trying out many ideas (to the point of using up departmental computing resources [2]).

They should also learn how to use revelant software for their subfield [12].

When a new researcher begins his or her own research, some tips to keep in mind are to dig deep, expect the unexpected, and to not get discouraged when failures inevitably occur. Failures can be mitigated by changing one's problem-solving approach [10].

3.4 Tools Needed in CS Research

The following is a list of the tools used by UTSA's research professors.

Useful tools for reading papers include the Internet/Google, PDF markup software, the UTSA physical library, and digital libraries including ACM, IEEE, Google Scholar, conference and pub locator web portals, and Springer.

Useful tools for drawing diagrams include Xfig and Gnuplot.

Useful tools for writing papers include LaTeX/bibtex, Endnote.

Useful tools for handling data include a good statistics textbook, public data sets, Matlab.

Useful tools for developing code include Python and similar scripting language, compilers, debuggers, Linux, and version control software.

Powerpoint is a useful tool for making presentations at conferences.

Subfield-specific tools mentioned include Waka for machine learning, Maxima (algebra software), Xen hypervisor software, and VMWare.

Almost all researchers agree that the best way to learn how to use a tool is through practice and experience. According to Dr. Robbins, another good way to learn how to use a tool is to teach someone else how to use it.

3.5 The Qualities of a Good Researcher

A good researcher has many qualities, including passion, patience, persistence [13], good attitude, perservering, good work ethic, eager for challenge, can face obstacles, [10].

A good reasearcher gains related knowledge, is a critical thinker, and has done well in his or her courses [10].

In order to be productive and efficient, he or she must also be good at context switching and time management [8].

If someone lacks some of these qualities but wants to become a researcher, he or she should work to give him or herself these qualites [10].

A good researcher should also have good managerial and leadership skills. They should know how and when to make presentations, how to get funding, how to interect with research communications, and how to support students [9].

4. CONCLUSIONS

This paper has shown the nature of research in general, computer science research specifically, and some of the things you will need to know to become a good computer science researcher.

5. REFERENCES

[1] D. Akopian. Personal Communication.

[2] T. Bylander. Personal Communication.

[3] S. Dey. A beginner's guide to computer science research. *XRDS*, 20(4), 2014.

[4] M. Gibson. Personal Communication.

[5] D. L. L. R. M. d. M. Jacques Wainer, Claudia G. Novoa Barsottini. Empirical evaluation in computer science research published by acm. *Information and Software Technology*, 51:1081–1085, January 2009.

[6] P. C. Jose Maria Cavero, Belen Vela. Computer science research: more production, less productivity. *Scientometrics*.

[7] T. Korkmaz. Personal Communication.

[8] P. Lama. Personal Communication.

[9] H. Maynard. Personal Communication.

[10] A. A. Muzahid. Personal Communication.

[11] J. Niu. Personal Communication.

[12] K. Robbins. Personal Communication.

[13] R. Sandhu. Personal Communication.

[14] A. S. Tosun. Personal Communication.

[15] I. V. V. Ramesh, Robert L. Glass. Research in computer science: an empirical study. *The Journal of Systems and Software*, 70:165–176, 2004.

[16] J. von Ronne. Personal Communication.