How to do Research in the Navigators@LASIGE

Alysson Bessani

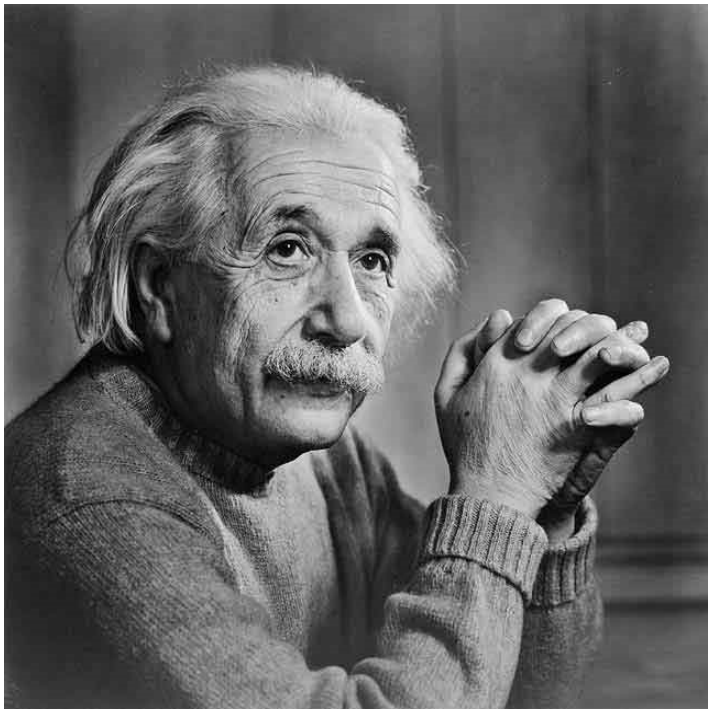
(with contributions from Antonio Casimiro, Paulo Verissimo)

Faculdade de Ciências da Universidade de Lisboa

https://navigators.di.fc.ul.pt/

https://lasige.pt/

Academia



Academia

(if you move in the first division)

- Highly competitive environment
  - Funding
  - Publishing
  - Impact
- Good researchers are high-competition athletes

Main issues

Define your Objectives
(Different Objectives at Different Levels)

Level	Publication	Quantity/yr
Undergrad	Meetings, soft pubs	1
Masters	Nat. Conf. A	1 or 2
	Nat. Journal A or Int. Conf. B	0 or 1
PhD/Pos-Doc	Int. Journal A or Conf. A	1+
	Int. Conf. B	1+

Read a lot!

- Which are the confs and journals on your field?
  - When you have the relevant list, go to the internet and read the title (and maybe the abstract) of most papers published there over the last ten years
  - Download the ones you think are interesting (in accordance with your advisor) and read them
  - Periodically, go to the conference/journal website to see what is new
- Find the key researchers and follow their work!

Read a lot!

- How many papers per week?
  - there are no magic figures, but, when you are starting, be prepared to, *on average*:
    - explore 5 to 10 per week (abstract, intro, concl.)
    - read 3 to 5 per week
  - this includes course assignments, your advisor suggestions, your initiative
  - it depends on the phase of your research

Read a lot!

(Don't worry if you don't understand everything)

- For each paper you read:
  - Ask yourself whether you understood it:
    - can you explain it *in your own words?*
  - Exercise your critical view!
    - Is the problem relevant?
    - Are assumptions realistic? Is the model sound?
    - What are the contributions? How practical the solution?
    - Is the provided evaluation/proof fair and/or rigorous?
    - Are experiments repeatable and comparable?
    - How could you improve this work?

Choosing a Research Topic

- Try to find a problem/topic that you care about…
  - Or, at least, find one whose importance you can explain
  - You NEED to know how to sell your idea as a worthwhile research topic:
    - to your advisor
    - to the Thesis Committee
    - to the community when you publish later

The Advisor(s)

- Your advisor will help you, but it is YOUR Masters/PhD
- It is your responsibility to make your advisor be excited about your work and work on it with you
  - The environment is informal, but we are professionals!
- **Golden rules to respect his/her time and effort :**
  - Be responsible with deadlines
    - Every deadline you miss, you lose the respect of your advisor
  - Be careful with the quality of what you deliver
    - Before delivering something to your advisor(s), ask yourself: *"Is this the best I can do (given the time constraints)?"*

Doing Research

The Idea

- Always ask the following questions:
  - What is the main contribution?
  - What are the benefits/merits (selling point) of the approach?
  - Why is it different from previous works?
- That's when you'll thank yourself for having read enough to answer these questions with some confidence

Formalization

- Problem definition
  - Define your problem and show why solving it is important
  - A solution in search of a problem is just the wrong way
- System model
  - Define your constraints and assumptions
  - You should characterize unambiguously both the problem and the environment where the proposed solution is valid
- Use the formalization approach that is understood and accepted in your community/field

Formalization

- Presenting the solution: Algorithm, Mechanism, Protocol
  - Intuition: give an intuitive overview of the solution
  - Self-containedness: choose the level of abstraction that fits the paper size
  - Pseudo-code: use good latex packages like algorithm2e to enhance presentation, use line numbers
- Formalizing the solution:
  - Operation: describe the operation of your solution concisely but precisely, referring to the pseudo-code (refer to line numbers)
  - Proofs: no protocol/algorithm is correct until proven so
  - Metrics: prototype or simulation may be useful ways of showing your point, whether you have made a proof

Implementation

- If your work requires implementation, try first to modify something that is already done/used
- Advantages:
  - Well-written (maybe) code but above all it's tested
  - You (automatically) gain a base for comparison
  - Makes the work more interesting for reviewers or thesis committee members
- Disadvantages:
  - Code from others is (generally) more complex than our toy examples and prototypes
  - The code may not work as expected

Evaluation

(Be honest and critic but don't be dumb!)

- Two attitudes to avoid
  - Being too smart: evaluate only the cases that you know are advantageous for your approach; ignore negative outliers
  - Being too critic: over-evaluate, -discuss and -justify the cases in which your approach is not the best one
- Common mistakes:
  - Not defining the questions that the evaluation aims to answer
  - Not giving enough detail so that experiment is reproducible
  - Not justifying experiment's parameters and workloads
  - Not comparing the proposed approach with others
  - Not interpreting, explaining and justifying obtained results

The Papers

Writing Papers

- Writing well is very hard!
  - First step to writing well is reading a lot
  - Then: practice, practice, practice
  - Every good paper is the result of many successive refinements
- Each paper has a *"champion"*
  - He/she is the owner of the paper, responsible for splitting the work among authors, asking for their parts and integrating the results in a single paper
  - Never work on a paper without a champion!

Writing Papers

- General philosophy:
  - Tell people about the problem you are going to solve
  - Tell people how you solve the problem
  - Tell them you solved it!

Writing Papers

- TODO list:
  - Description of the problem
  - Make contribution and significance clear
  - Related work
  - Describe environment and model
  - Describe the solution
  - Validate your solution
  - Lessons learned (Why is your paper worth reading?)

Writing Papers

- What writing a good scientific paper is about
  - it must: (i) not only be correct; but (ii) perceived as useful by the community; and (iii) interesting to read
  - papers with just (i) count for your curriculum but they are *write-only* papers, i.e., papers that no one reads, *ergo* no one cites
  - papers with (i) and (ii) are ok, specially for Calvinists
  - papers with all three, readers will: love you for that, cite you a lot more, be willing to read your next one

Writing Papers

- Steps to writing a paper:
  1. Write the storyboard for yourself and other authors: a paper should be a good story
  2. Build a structure (sections and sub-sections)
  3. Each section must be filled with a bulleted list
     - *You are telling a story, each argument needs to be linked…*
     - *A scientific text is an algorithm in itself (hence LaTEX☺) !*
  4. Add figures, tables, and informal references
  5. Consolidate bullets into paragraphs
  6. Collect formal references and related work
  7. Reiterate by successive refinement until done

Writing Papers (wrap-up)

- The introduction needs to be **perfect**
  - Most reviewers can decide to reject your paper after reading the introduction
- Same for the presentation and style (text, figures and general appearance)
  - Remember, we don't do write-only papers
- Ask for feedback from your colleagues
  - Sometimes better if some don't work in the same area (like reviewers); *feedback* is fundamental!
  - Include a couple of outside experts

Submitting Papers

- Workshops
  - Very good for
    - disseminating early results
    - discussing a problem
    - getting feedback
    - meeting other people working on your area
  - Counts little for CV evaluation
  - Some of them are very good (and competitive): HotOS, HotNet & HotStorage

Submitting Papers

- Conferences
  - The good conferences in CSE may be harder and have more prestige than the best journals
    - TYP acceptance rate less than 20%
    - Papers with 12-16 pages (longer than some top journals)
  - These are what we call *heavy-weight conferences*
  - PCs in each community expect a particular style of papers, so before submitting to a top conference, try to learn their style (i.e., read a lot!)

Submitting Papers

- Some Excellent and VG confs (not complete):
  - Distributed Systems: **ICDCS**, IPDPS, Middleware
  - Distributed Syst. Theory: **PODC, DISC**, OPODIS
  - Dependability: **DSN**, SRDS, ISSRE
  - Security: **Security & Privacy (Oakland), CCS, USENIX Security**, NDSS, Crypto
  - Networks: **SIGCOMM, NSDI**, INFOCOM, CoNEXT
  - Systems: **SOSP/OSDI**, EuroSys, USENIX ATC
  - Real-time: **RTSS, RTAS**, EuroMicro
  - Machine Learning: **NIPS, ICML**, ECML-PKDD

Submitting Papers

- Acceptance rate
  - A good half of the papers submitted to a top conference *don't stand a chance* even before the PC show starts
  - From the remainder, bottom half have *little chances*
- If you follow the rules presented, you have:
  - a good chance of staying out of the sudden-death half, right from the beginning
  - Getting to the top quarter and fighting for an accept is another thing…

Submitting Papers

- As you build experience, you should aim to *systematically* be in the top 25%
  - You get to know you're there because reviews get better
  - Getting there implies keeping on reading reviews with self-criticism and scrupulously analysing constructive criticism
  - Above a certain standard, fair English is an obstacle --- not making mistakes is not enough, you need style.
    - Improve!
    - Rely on senior co-authors, their touch may make the difference
- Still, you paper may be accepted or not ☺
  - Everyone has rejected papers! Especially if you aim high!

Submitting Papers

- Journal
  - Disadvantages:
    - “arguably” less immediate visibility, which may be counterproductive in a lively field as CSE
    - to overcome this, consider first submitting to conferences and evolve best works to journal
  - Advantages:
    - Science bureaucrats love it, gives substance to your CV and plus it makes sense, it’s an archival grade work, read below

Submitting Papers

- Papers in the best journals are **substantive** and **archival grade**
  - Clear and complete contribution in a subject
  - Rigorous in the formalization, proofs or metrics
  - Carefully evaluated, no loose ends
- Reviewers are generally more responsible and accountable
  - You have a chance for a dialogue and rebuttal
  - (this is the case also in many top conferences)

Submitting Papers

- Revising and Responding to Reviewers
  - Always show that you took reviewers' comments into account, through the response letter
  - Consider politely challenging the review points with which you don't agree, the editor is an arbiter between you and the reviewer
  - A good method to prepare both your revision and your response, is to pass all reviews to a text processor and exhaustively comment all significant remarks *in-line in different colour*, proposing what to do to address or challenge

Submitting Papers

- Some good journals and magazines (far from complete) in no special order:
  - IEEE Transactions on …
  - ACM Transactions on …
  - Journal of ACM
  - Distributed Computing (Springer)
  - Journal of Real-Time Systems (Springer)
  - Computer Networks
  - IEEE Security and Privacy Magazine
  - Journal of Computer Security

The Reviewer

- Often (though not always) reviewers are very smart and have good intentions
- However,
  - They don’t have time
  - They expect fair amounts of scientific and engineering work
  - They may not be experts in your topic
  - Some (rare) may not have good intentions

The Reviewer

- Keep these things in mind:
  - Don’t make it easy for them to reject your paper
  - Try to finish it up as sphere (no place to grab)
  - Citations are (typically) free, certain people don’t like not to be cited
  - Don’t belittle past work that you are advancing from:
    - you should step on others’ shoulders, not on their toes
    - you may be next…

Navigators' Publishing Policy

- Submit preliminary work early to a *good* workshop
- Submit a finished paper to a *good* conference
- If accepted, great!
  - If it is worthwhile, prepare an extended version (at least 25% of new content) and submit to a journal
- If rejected, ask yourself:
  - Some problems or just unlucky? Solve them and try again
  - Misunderstood? Under fire? Improve and send to a journal

To Conclude…

- What you get for staying in the academia:
  - You don’t need to work under direct orders
  - You get to participate in defining what you work on
  - You get to know the world and meet the smartest people
  - You have substantial freedom to manage your time
- What you must give:
  - Reciprocate with top quality, self-responsibility, team spirit
  - **Work hard! Be better than you were yesterday!**
  - Love what you do and be proud of how good you are
  - Don’t be afraid to have ideas, ask questions, criticize
  - Be your greatest critic but accept constructive criticism

Some References

- Randy Pausch. **The Last Lecture**.
  - Vídeo: *http://www.youtube.com/watch?v=ji5_MqicxSo*
- Strunk & White. **The Elements of Style**. 4th edition. Longman.
  - Book about the style of written English, highly recommended.
- Phil Guo. **The PhD Grind - A Ph.D. Student Memoir**. 2012.
  - http://pgbovine.net/PhD-memoir/pguo-PhD-grind.pdf
- Levin & Redell. **How (and how not) to write a good systems paper.**
  - http://www.usenix.org/event/samples/submit/advice.html
- John Wilkes. **How to write a good [systems] paper (things I wish my mother had told me)**, *EuroSys 2006 Authoring Workshop*
  - http://www.cs.kuleuven.be/conference/EuroSys2006/authoring_workshop/wilkes.pdf
- Material in the pages of professors Priya Narasimhan and Mike Dahlin
  - http://www.cs.cmu.edu/~priya/tech-writing.html
  - http://www.cs.utexas.edu/users/dahlin/advice.html