# Writing Research Papers

**Lei Yu**
**Graduate Skills Seminar, 2024**
**Ack: Prof. Stacy Patterson and Prof. Lirong Xia**

**Credit to Simon Peyton Jones, How to write a great research paper, https://www.microsoft.com/en-us/research/academic-program/write-great-research-paper/**

# Why write papers?

- Purpose of writing a research paper
  - To impress others and get recognition? NO
  - To tell people what you have done? NO
  - Share something new? NO

**To Convey a useful and re-usable idea**

# Why write papers?

- Papers Communicate Ideas
- You have a good idea you want to tell others about
  - Your research community, your advisor, your thesis committee, industry leaders, government agencies, …
- The great ideas are worthless if you keep them to yourself
- Papers are far more durable than programs
  - Think Mozart

# Why write papers?

Benefits:

- You can get feedback from your research community (workshop paper, vision paper, conference paper)
- You can learn more about a topic (survey paper)
- You need to publish papers to graduate, get a good job, and get a promotion, …
  - Publish or perish
- Writing can help you with your research

# Getting ahead entails good ideas communicated well

- **Success in research** = good results + good communication about your results
- **Success in business** = good ideas + good follow-through + good communication about your work
- **Good communication** =
  - “Elevator pitch” for professional conversations
  - Good presentations
  - Good writing
    - Conferences, journals, proposals, reports, white papers, etc.

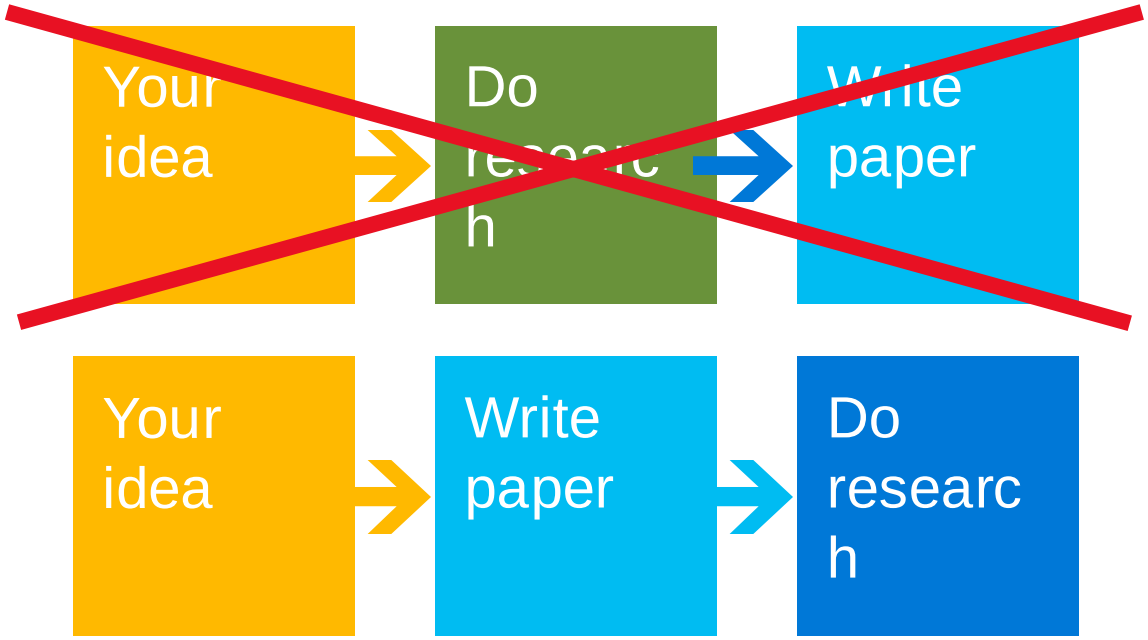
# How to write a good research paper

- 1. Don’t Wait, Don’t Procrastinate: Write

# Writing papers: model 1


Credit: Simon Peyton Jones

# Writing papers: model 2



# Writing as a Creative Process

***Writing advances your technical understanding.*** *Many details and the overall story are refined when you write things up.*

- **Writing can help organize the way you think about your results**
  - You may have created the results in an exploratory way, but you should write them up as part of a well-structured plan
- **Writing can help put your work in context**
  - You'll discover a lot about your work by setting it in context with others
- **Writing can help you discover missing pieces and mistakes**
  - There is a big difference between a proof sketch and a proof
- **Writing may lead you to additional work**
  - Do you need to do more in order to present strong results?

# Writing papers: model 2


- Forces us to be clear, focused
- Crystallises what we don't understand
- Opens the way to dialogue with others: reality check, critique, and collaboration

# Writing papers: model 2


Writing papers is a primary mechanism for doing research (not just for reporting it)

# Preliminaries before Writing

- *Identify your key idea*
  - Your initial idea may seem insignificant to you. Do not be intimidated. It does not have to be a fantastic idea before writing
  - Writing the paper is how you develop the idea in the first place
- By the time you finish writing,
  - Your paper should have just one “ping”: one clear, sharp idea
  - Many papers contain good ideas, but do not distil what they are.
  - Make certain that the reader is in no doubt what the idea is. Be 100% explicit:
    - “The main idea of this paper is....”
    - “In this section we present the main contributions of the paper.”

# Preliminaries before Writing

- ***Review and Renew Your Literature Search***

- ***Understand the cultural context and Who Your Audience Is.***
  - ***Writing exists in a cultural context.*** *Know your professional culture and how it communicates. Your writing should be in the same style as others in your group and should be* ***optimized for your reader.***
  - Depends on the conference or journal
  - Researchers who are experts in your area and adjacent areas
  - People who want to learn more about you and/or your ideas

  - **But before that, your readers are the reviewers for the conference or journal**

# Writing

- ***Tell a story***

Your narrative flow

Imagine you are explaining at a whiteboard

- Here is a problem
- It's an interesting problem
- It's an unsolved problem
- Here is my idea
- My idea works (details, data)
- Here's how my idea compares to other people's approaches

# Conference Paper Structure

- The structure of your paper is dependent on conference and area. **Read "best papers" from your target conference as a model.** (Best papers in many conferences at http://jeffhuang.com/best_paper_awards.html)
- **Generic format**
  - Title
  - Abstract
  - Introduction
  - (Related work)
  - Problem Description
  - Solution/Method
  - Results
  - (Related work)
  - Conclusion
  - References

# Conference Paper Structure – More Detail 1

## 1. Abstract

- 1-2 paragraphs summarizing the paper and results.
  Should be **well-written and broadly understandable**.
  Typically, does **not** include motivation for the work.

## 2. Introduction

- ~1 page – should tell reviewer everything they need to know about what you are doing and why it is important.
- **This is the time to catch the reader and make a positive first impression. Spend the extra time to make this compelling, exciting and interesting.**

## 3. Related work

- **Generously include work that's relevant, especially from likely reviewers**
- Diplomatically describe why their work doesn't solve your problem ("Berman's work focused on networks in which communication costs are zero. Our work focuses on networks in which communications costs are positive and vary dynamically")
- Related work is sometimes positioned later in the paper.

# Conference Paper Structure – More Detail 2

4. **Problem Description**
   - Clearly describe the problem you're solving.
5. **Solution**
   - Provide enough detail to make your approach **credible** and **reproducible.**
   - Give pseudocode if applicable.
6. **Results (Theoretical)**
   - Give formal statements of theoretical results.
   - Provide proofs or pointers to proofs.
   - Discuss significance and implications of the theory.
7. **Results (Experimental)**
   - Make work visually accessible and interesting. Use well-described graphs and visuals (tables, etc.). **Make sure captions are self-contained.**
   - Provide insight into trends in your data.

# Conference Paper Structure – More Detail 3

8. **(Future Work)**
   - What needs to be done next?
   - This can help increase the interest and motivation for the work you've already done.
   - Sometimes part of the conclusion.
9. **Conclusion**
   - .5 page? Should be as self-contained as possible, compelling, and less than a page: What did you do, how did you do it, what are the results, why should we care?
10. **Acknowledgements**
    - People who helped who are not authors. Be generous. Include funding sources.
11. **References**
    - Be relevant and generous. Use the standardized reference formats for your community.
    - Be careful with self-citation. Don't overdo it.

# Complete a draft

- ***Begin with the "idea" part – technical section, experiment section***
  - ***It is the easy part to write based on your idea***
- ***Create the outline***
  - ***technical approach, experiment design, the subproblems you need to investigate***
- ***Do not start with the introduction.***
  - ***Hardest part of the paper***
  - ***Without clear conclusions, it could be a waste of time.***
- ***Resist the temptation to correct and edit as you go***
  - ***Produce a complete first draft not a perfect draft***

# Submission

- **Audience:** Specialists in your general area (e.g., programming languages, data mining, HPC)
  - In peer-reviewed conferences, first you need to get accepted by the **Review Committee**
- **Review committees:**
  - **Have a lot of papers to read in a short period of time -- You need to "sell them" early**
    - Make your results compelling and clear on the first page
    - Make your paper easy to skim (graphics, organization, line spacing, font)
  - **Are looking for new results that advance the state of the art**
    - Make the innovative aspects of your paper clear up front
  - **Are often looking for "hot" topics**
    - Can you relate your results to a community / national priority or new area?

# The introduction (1 page)

- Describe the problem
- State your contributions

...and that is all

ONE PAGE!

# Describe the problem

## 1 Introduction

There are two basic ways to implement function application in a higher-order language, when the function is unknown: the *push/enter* model or the *eval/apply* model [11]. To illustrate the difference, consider the higher-order function **zipWith**, which zips together two lists, using a function **k** to combine corresponding list elements:

```
zipWith :: (a->b->c) -> [a] -> [b] -> [c]
zipWith k []     []     = []
zipWith k (x:xs) (y:ys) = k x y : zipWith xs ys
```

Here **k** is an *unknown function*, passed as an argument; global flow analysis aside, the compiler does not know what function **k** is bound to. How should the compiler deal with the call **k x y** in the body of **zipWith**? It can't blithely apply **k** to two arguments, because **k** might in reality take just one argument and compute for a while before returning a function that consumes the next argument; or **k** might take three arguments, so that the result of the **zipWith** is a list of functions.

Use an example to introduce the problem

# Molehills not mountains

Example: “Computer programs often have bugs. It is very important to eliminate these bugs [1,2]. Many researchers have tried [3,4,5,6]. It really is very important.”

Yawn!

Example: “Consider this program, which has an interesting bug. <brief description>. We will show an automatic technique for identifying and removing such bugs”

Cool!

# State your contributions

- Write the list of contributions first
- The list of contributions drives the entire paper: the paper substantiates the claims you have made
- Reader thinks “gosh, if they can really deliver this, that’s be exciting; I’d better read on”

# State your contributions

Do not leave the reader to guess what your contributions are!

Which of the two is best in practice? The trouble is that the evaluation model has a pervasive effect on the implementation, so it is too much work to implement both and pick the best. Historically, compilers for strict languages (using call-by-value) have tended to use eval/apply, while those for lazy languages (using call-by-need) have often used push/enter, but this is 90% historical accident — either approach will work in both settings. In practice, implementors choose one of the two approaches based on a qualitative assessment of the trade-offs. In this paper we put the choice on a firmer basis:

- We explain precisely what the two models are, in a common notational framework (Section 4). Surprisingly, this has not been done before.
- The choice of evaluation model affects many other design choices in subtle but pervasive ways. We identify and discuss these effects in Sections 5 and 6, and contrast them in Section 7. There are lots of nitty-gritty details here, for which we make no apology — they were far from obvious to us, and articulating these details is one of our main contributions.

In terms of its impact on compiler and run-time system complexity, eval/apply seems decisively superior, principally because push/enter requires a stack like no other: stack-walking

Bulleted list of contributions

# Contributions should be refutable

| No! | Yes! |
| --- | --- |
| We describe the WizWoz system. It is really cool. | We give the syntax and semantics of a language that supports concurrent processes (Section 3). Its innovative features are... |
| We study its properties | We prove that the type system is sound, and that type checking is decidable (Section 4) |
| We have used WizWoz in practice | We have built a GUI toolkit in WizWoz, and used it to implement a text editor (Section 5). The result is half the length of the Java version. |

# Evidence

- Your introduction makes claims
- The body of the paper provides evidence to support each claim
- Check each claim in the introduction, identify the evidence, and forward-reference it from the claim
- “Evidence” can be: analysis and comparison, theorems, measurements, case studies

# No “rest of this paper is...”

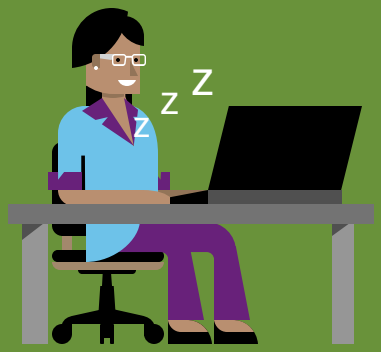
- Not:
  “The rest of this paper is structured as follows. Section 2 introduces the problem. Section 3 ...Finally, Section 8 concludes”.
- Instead, use forward references from the narrative in the introduction. The introduction (including the contributions) should survey the whole paper, and therefore forward reference every important part.

# Structure

- Abstract (4 sentences)
- Introduction (1 page)
- Related work
- The problem (1 page)
- My idea (2 pages)
- The details (5 pages)
- Conclusions and further work (0.5 pages)

NO!

# No related work yet!



- Problem 1: the reader knows nothing about the problem yet; so your (highly compressed) description of various technical tradeoffs is absolutely incomprehensible
- Problem 2: describing alternative approaches gets between the reader and your idea

# Structure

- Abstract (4 sentences)
- Introduction (1 page)
- The problem (1 page)
- My idea (2 pages)
- The details (5 pages)
- Related work (1-2 pages)
- Conclusions and further work (0.5 pages)

YES!

# Writing Journal Papers

- **Journal papers are the archival record of your work. This is the place for comprehensive detail and thorough description of what you have done.**
- **Audience:** Specialists in your specific area (e.g. exascale programming environments) and in the general area (e.g. HPC)
- **A journal paper** can be based on a conference paper, but should provide some new content
  - The amount of new content depends on the journal
  - In my field(s), the requirements for new content are increasing

# Journal Paper Structure

- Organization of the paper is dependent on journal and area. **Read other papers from the journal as a guide.**
- **Generic format**
  - Introduction
  - (Related work)
  - System Model/Problem Description
  - Proposed Solution
  - Results
  - (Related work)
  - Conclusion
  - **Appendices**
  - References

# Journal Paper Structure – More Detail 1

- **Use the same basic structure for journal and conference papers.** All guidelines about clear and compelling writing in conference papers apply.
- **In a Journal paper, you have more room and are expected to go into greater detail.**
  - **Reviewers expect to spend more time on these and provide detailed feedback to you.**
  - Write the paper assuming that every detail will be gone through with a fine-toothed comb.
- **Differences between Journal and Conference Papers**
  - **Introduction** -- you have a bit more room than a page. In a 20 page paper, you can spend 2+ pages motivating the problem and your solution.
  - **Related work** – You can go more into detail about what other people have done and where your work fits in. Remember that you need to **both show that your work is different and not put down other's research**

# Journal Paper Structure – More Detail 2

- **Differences between Journal and Conference Papers:**
  - **Approach** – Provide a detailed methodology that could be reproduced. Provide enough detail about parameters, hardware, software, data, versions **so that someone else could reproduce your results** with the same setup if possible.
  - **Results** – **Explain your results and their significance thoroughly.** Why did the graph spike? Under what circumstances was one method better than another?
  - **Future Work** – Include future approaches and problems that this work could support. Spend < 1 page on this.
  - **Conclusion** – Make this self-contained and clear. Spend < 1 page on this.
  - **References** – Can include even more refs. Be generous and thorough.

# The Paper Review Process

- *"Peer review is a process in which a scientific paper is evaluated by a group of experts in the same field to make sure it meets the necessary standards for acceptance and publication"**

- For conferences
  - The conference organizers select a Technical Program Committee (TPC) who are experts in the conference topics
  - Each paper is reviewed by 3-4 TPC members
    - They review the paper and provide comments and criticisms
  - The TPC Chairs use the reviews to decide which papers are accepted.
  - Process usually takes a few months
  - Some conferences allow a rebuttal or single revision

*https://conferences.ieeeauthorcenter.ieee.org/understand-peer-review/

# The Paper Review Process – Cont.

- For journals
  - Your paper is assigned to an Associate Editor (AE)
  - The AE sends your paper to 3-4 experts
    - Often selected from the papers in your Related Work section
    - They review the paper and provide comments and criticisms
  - The AE uses the reviews to make a recommendation for the paper
    - Accept, Minor Revision, Major Revision, or Reject
    - Decision should take 4 to 6 months, but can be much longer
  - If the recommendation is Major or Minor Revision, you revise the paper according to the reviews and resubmit
    - Usually given 3 months to make revisions
  - Revision assigned to the same set of reviewers
    - They review paper again
  - AE makes decision based on these reviews - usually accept or reject.
    - Some journals allow multiple revisions

# A few words word about reviewers….

- Most reviewers want to do a good job
  - Read and evaluate paper thoroughly and fairly
  - Provide useful feedback
  - Make a thoughtful decision/recommendation
- But reviewers are very busy people
  - When I review a paper, I usually only have time to read it once
- And there is randomness in the process

- So, authors should make things as easy on the reviewers as possible
  - Fortunately, this is the same as writing a good paper

# Optimizing your chances of success (conference and journal papers) 1

- **Make the paper easy / fun to read**
  - **Use graphics** to compellingly convey the message
  - Use font / spacing / italics and bolding / section organization / color to make the paper easy to read
    - **Don't jam every possible space and use super-small font**
  - **Do multiple drafts** until you get it to a level that will promote your success
- **Present your work professionally**
  - **Use "we"** vs. "I", even if there is only one of you
  - **Don't oversell nor undersell.** Study how other authors subtly promote the importance of their work without sounding arrogant (the "humblebrag"!).
  - **Look at "Best Paper" recipients** in your conference or highly ranked journal articles for good examples of writing and results
  - **Do not plagiarize** – if you want to refer to something explicitly or implicitly, cite the work and give credit to the authors.

# Optimizing your chances of success 2

- **Strive to make your work reproducible:**
  - Include enough detail about methods, hardware, software, parameters, etc. so that someone could conceivably reproduce your results.
  - Share code and data when possible. Issues about what to share and what is competitive advantage still under discussion https://web.stanford.edu/~vcs/papers/RoundtableDeclaration2010.pdf
- **Use "Red-team" reviewers:**
  - Pick knowledgeable and trustworthy colleagues to read and critique your paper.
  - **Revise the paper** based on their reactions and their comments on strengths and weaknesses.

# Optimizing your chances of success 3

- **Be a nice citizen**
  - Pay attention to page length. If there are page limits (e.g. in conferences), don't just make things smaller to fit in.
  - Be generous with related work – include relevant potential reviewers
  - Don't dis others
- **Pay attention to the deadlines**
  - Sometimes there is an abstract deadline before the conference paper deadline
  - Sometimes deadlines are extended, but don't count on this
  - For conferences, submit early and submit often
- **Proofread, proofread, proofread!**

# What to do if your paper is rejected?

- Rejection is no fun, but it happens to all of us.
- Sometimes reviews seem unfair or incorrect.
  - The Reviewer 2 meme exists for a reason
  - Need to make the best of it
  - Can report blatant misconduct
- If the paper is not accepted… **Don’t give up**
  - Look at the reviews and improve the work
  - Get more results if needed
  - Ask for help from your network
  - Send it somewhere else (or for another round)

# Listening to your reviewers

Treat every review like gold dust

Be (truly) grateful for criticism as well as praise

This is really, really, really hard

But it's really, really, really, really, really, really, really, really, really, really, really important

# Listening to your reviewers

- Read every criticism as a positive suggestion for something you could explain more clearly
- DO NOT respond “you stupid person, I meant X”.
- INSTEAD: fix the paper so that X is apparent even to the stupidest reader.
- Thank them warmly.  They have given up their time for you.

# Resources

https://www.microsoft.com/en-us/research/academic-program/write-great-research-paper/

https://www.youtube.com/watch?v=UY7sVKJPTMA&t=10s&ab_channel=SurvivingandThrivinginHigherEducation

*http://www.cs.ucr.edu/~eamonn/Keogh_SIGKDD09_tutorial.pdf*