Guidelines for Writing a Paper

Performance Engineering Seminar
Jan-Patrick Lehr, Yannic Fischler, Michael Burger
10/05/2022

1 Introduction

This document is meant to give an idea of what we look for in papers. It is to be seen as a – hopefully – helpful guide with practical hints for everybody. Most suggestions are general, but some of the advice is LaTeX-specific. This document, together with the HowTo-Review guide, shall give a good indication of how to approach a good paper.

2 General parts contained in a paper

Aside the content of the paper, it is **essential** to use the correct template. For PEng we provide a modified TU Darmstadt template, which is to be used.

We encourage to follow this structure in your paper. One degree of freedom that may fit your style of writing is to put the related work after the discussion to set the related work into context of what you did.

- **Abstract:** Briefly summarize the whole paper. In particular, its contributions. *Hint:* Mention a specific result to show the impact of the work.

- **Introduction:** Motivate and pose the scientific question addressed in the paper.

- **Related work:** Reference work that has been done by other researchers in the field and that you see as related.

- **Main part:** Explain what you did, why you did it and how you did it. (cues: implementation, methodology, evaluation)

- **Conclusion:** Draw conclusion and how your work answers the question and/or which open questions remain.

- **Future work:** Briefly outline some research ideas that can be pursued.

3 Outline and structure

- In this work, you should orient at our structure given above.

- Neither be too fine granular (e.g., creating sub-sub-sub-sections with two sentences of content each) nor too coarse grained (only two sections for three pages of text. You can also omit the outline.

- Your paper should not have a table of contents.

- You should not have a list of figures, list of tables or list of abbreviations in a paper.

- Use short and meaningful section titles. For example, not "How the code scales for a varying number of parallel threads employed" but "Scaling analysis"

- For a paper it looks better if not each section starts with a new page.

- Do not exceed the page limit.

- Do not exceed the text width (latex warning: overfull hbox). In general, follow warnings in latex. Most of them are useful to improve the quality of your paper.

4 General writing style

- Use a spell checker and proof-read your draft!

- Use short sentences instead of multi-clause sentences.

- No narrative style.

- No superlatives, no vague adjectives.

- No rhetorical questions.

- Do not use "Füllwörter" or even "Füllsätze". Formulate short and precise.

- Avoid I/me-constructions.

- No subjective statements, emotions and valuations.

- Do not claim something that you cannot proof with your own data or a scientific reference (e.g., "GPUs are known to be faster than CPUs").

- Use a level of abstraction which allows a non-expert to understand at least the crucial points of your work. In your case: Every Bachelor-student of computer science should be able to get what you have done.

- Write out abbreviations which are not common (common are, for example, CPU and OS) in the domain of the target audience one time at their first appearance in the text.

- Briefly explain terms which are not common and if they are very important for your statement/work, you should add a reference with further information.

- State open questions and do not hide them.

- Be aware of automatic line breaks, especially in proper names or between related "words", e.g. Figure 1. You can use a "stronger binding blank space" using a~ instead of a space character.

- Be consistent, keep your style of writing, e.g. use Figure or Fig. and don't mix it.

5 Figures, tables and listings

- Use figures where they make sense not to improve the "optics". For example, a picture of a Haswell processor that you are employing as test system provides no information.

- Figures have to be referenced in the text. When referencing a specific figure, capitalize it, e.g., Figure 1.

- The caption should clarify what the figure is about.

- Avoid doubling of content in text and figures. Text explains or interprets the figure but does not repeat its content.

- Be careful how many information you put into a figure / table. Too little information or trivia may "scare" the reader as well as overloading the figure.

- If you copy figures then give the source in the caption.

- If you have an inspiration/template for your figure also mention it.

- Figures must be readable and all details must be perceptible also in the printed version.

- Try to avoid the use of pixel graphics. If you need them use high resolution graphics. So no screenshots of a 300x300 diagram (redraw it instead).

- If you employ screenshots, limit them to the interesting part. For example, no full screen shot of the Intel Vtune Amplifier if you want to show the four hotspots.

- Do not use huge tables with measurement data. Condense to the important information (do not give each runtime for 50 repetitions with the same configuration).

- If you present numbers: Present only up to relevant digits, e.g., 40.123423% should instead be 40.1% or even 40%, depending on the context.

- Neither figures nor tables should be wider than the text width.

- Listings can be very helpful for some situations, and may be superfluous in others. If you want to highlight a particular detail, consider using a listing. They should, however, be short and concise to highlight the **key aspect**, i.e., use pseudo code or comments to communicate context like variable initialization and alike.

6 Bibliography

- Wikipedia is not a scientific reference.

- Be also very careful when citing homepages even from scientific institutions. If you cite websites you have to add the date of last access.

- Only list literature that you have really used.

- When you use literature, also check that their content makes sense!

- Be specific: Give the page number you are referring to (in particular for books!).

- Always do literature research for related work.

- Good starting points are for example scholar.google.com or basic papers/books about your topic which contain more specific references or have been cited by later work on that topic.

- Try to find recent literature.

- You can use the Citavi campus license for literature management as a good training for Bachelor's-/Master's thesis, or you can use JabRef as another literature management system.

- Include the reference in your text, e.g. "Musterfrau et al. [4] show that...". Do not just put the reference at the end of a text block.

- Do not include superfluous cites, like citing GCC or OpenMP.

- If you want to cite a software try to find the paper in which the code was first published. Most developers give you information on how to cite their software.