— Your Project Title —

John Doe
School of Computer Science
Carleton University
Ottawa, Canada K1S 5B6
John-Doe@scs.carleton.ca

October 4, 2021

Abstract

A very concise summary of the problem addressed and solution presented in this paper.

1 Introduction

Introduce your project topic (start from parallel computing in general and lead to your particular topic). Describe your project goals. Describe what you have achieved in your project. Outline the structure of your paper: In Section 2, we will review the relevant literature. ... Section 6 concludes the paper.

2 Literature Review

Give an overview of the relevant literature. Cite all relevant papers, like [2], [7], [3], [6], [5], [1], and [4]. Outline for each paper the relevant results in relation to your project. Make sure that you don't just list all relevant papers in random order. Devise a scheme to group papers by subject. The goal of this section is to present to the reader the state-of-the-art in the field selected for your project.

3 Problem Statement

1. A concise statement of the question that your paper tackles.
2. Justification, by direct reference to your Literature Review, that your question is previously unanswered.
3. Discussion of why it is worthwhile to answer this question.

4 Proposed Solution: ...

This part of the paper is much more free-form. It may have one or several sections and subsections. But it all has only one purpose: to convince the reader that you answered the question or solved the problem that you set for yourself. In this section you will for example present new algorithms you developed and your implementation of these new algorithms.

Figure 1 is an example of a drawing created with *mdraw* or *epsfig*.

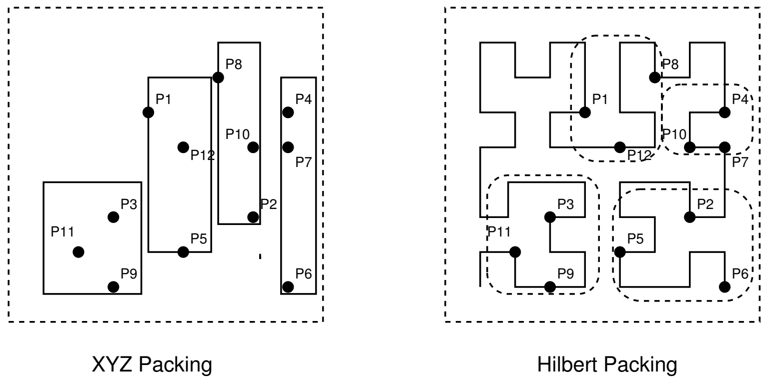

Figure 1: XYZ and Hilbert Packings

5 Experimental Evaluation

This section is not mandatory for all papers (for example theory papers) but typically required for papers in the field of parallel computing. After all, parallel computing is all about compute performance. Here you present performance data obtained from e.g. running your newly developed algorithms and code on a parallel machine using some benchmark input data. Typically, you need to describe the parallel machine you used and the data that you used as input. The main results are usually performance graphs, typically speedup curves. You want to evaluate your code on different input data sets highlighting the strengths and weaknesses of your code. Don't just use best case scenarios. People will call you on that. Discuss the results obtained.

Figure 2 is a typical example of an experimental evaluation result. Such graphs are ususally created with GnuPlot.

6 Conclusions

You generally cover three things in the Conclusions section.

1. Conclusions
2. Summary of Contributions
3. Future Research

Conclusions are not a rambling summary of the thesis: they are short, concise statements of the inferences that you have made because of your work. All conclusions should be directly related to the research question.

The Summary of Contributions will be much sought and carefully read by the readers. Here you list the contributions of new knowledge that your paper makes. Of course, the

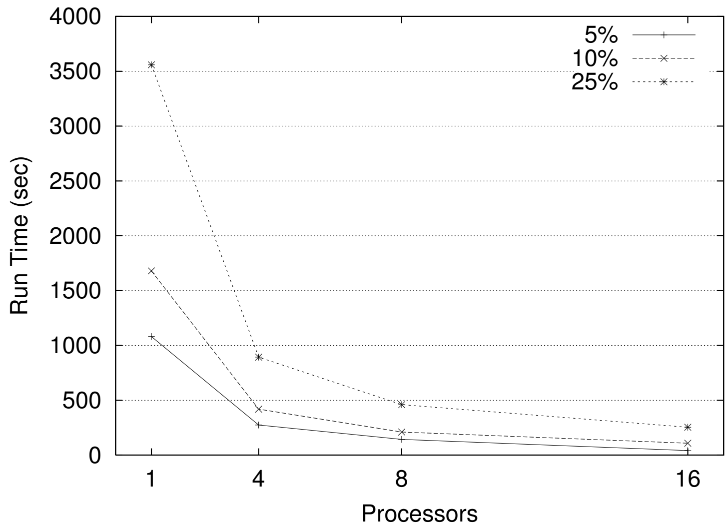

Figure 2: Measured Running Times Of Some Unknown Algorithm Implementation

paper itself must substantiate any claims made here. There is often some overlap with the Conclusions, but that's okay.

The Future Research should indicate interesting new problems arising from your work. No paper ever solves everything. In fact, the best research papers lead to new research questions for other researchers to work on.

References

[1] Y. Chen, F. Dehne, T. Eavis, D. Green, A. Rau-Chaplin, and E. Sithirasenan. cgmOLAP: Efficient parallel generation and querying of terabyte size ROLAP data cubes. In *Proc. 22nd Int. Conf. on Data Engineering (ICDE)*, pages 164–164. IEEE Comp. Soc. Dig. Library, 2006.

[2] F. Dehne, T. Eavis, and B. Liang. Compressing data cubes in parallel OLAP systems. *Data Science Journal*, 6:S184–S197, 2007.

[3] F. Dehne, T. Eavis, and A. Rau-Chaplin. The cgmCUBE project: Optimizing parallel data cube generation for ROLAP. *Distributed and Parallel Databases*, 19(1):29–62, 2006.

[4] F. Dehne, M. Fellows, M. Langston, F. Rosamond, and K. Stevens. An $o(2^{O(k)}n^3)$ FPT algorithm for the undirected feedback vertex set problem. In *Proc. 11th Int. Computing and Combinatorics Conf. (COCOON)*, pages 859–869. Springer LNCS 3595, 2005.

[5] F. Dehne, M. Langston, X. Luo, S. Pitre, P. Shaw, and Y. Zhang. The cluster editing problem: Implementations and experiments. In *Proc. Int. Workshop on Parameterized and Exact Computation (IWPEC)*, pages 13–24. Springer LNCS 4169, 2006.

[6] M. Lawrence, F. Dehne, and A. Rau-Chaplin. Implementing OLAP query fragment aggregation and recombination for the OLAP enabled grid. In *Proc. International Parallel and Distributed Processing Symposium (IPDPS), High-Performance Grid Computing Workshop*, pages 1–8. IEEE Comp. Soc. Dig. Library, 2007.

[7] S. Pitre, F. Dehne, A. Chan, J. Cheetham, A. Duong, A. Emili, M. Gebbia, J. Greenblatt, M. Jessulat, N. Krogan, X. Luo, and A. Golshani. PIPE: a protein-protein interaction prediction engine based on the re-occurring short polypeptide sequences between known interacting protein pairs. *BMC Bioinformatics*, 7:365 (15 pages), 2006, available via PubMed at http://www.biomedcentral.com/pubmed/16872538.