Haskell versus Rust: Benchmarking Efficient Concurrent Pathfinding in a Purely-Functional Programming Language

Prince Bernie B. Colis
Bachelor of Science in Computer Science
John Kenneth S. Lesaba
Bachelor of Science in Computer Science
Jon Ariel N. Maravilla
Bachelor of Science in Computer Science
Karl Frederick R. Roldan
Bachelor of Science in Computer Science

Senior project submitted to the faculty of the
Department of Computer Science
College of Computer Studies, Ateneo de Naga University
in partial fulfillment of the requirements for their respective
Bachelor of Science degrees

Project Advisor: Adrian Leo Pajarillo
First Panel Member
Second Panel Member
Third Panel Member

Month Day, 2020
Naga City, Philippines

Keywords: parallel programming, functional programming, graph theory

The Senior Project entitled

Haskell versus Rust: Benchmarking Efficient Concurrent Pathfinding in a Purely-Functional Programming Language

developed by

Prince Bernie B. Colis
Bachelor of Science in Computer Science

John Kenneth S. Lesaba
Bachelor of Science in Computer Science

Jon Ariel N. Maravilla
Bachelor of Science in Computer Science

Karl Frederick R. Roldan
Bachelor of Science in Computer Science

and submitted in partial fulfillment of the requirements of their respective Bachelor of Science degrees has been rigorously examined and recommended for approval and acceptance.

First Panel Member
Panel Member
Date signed: _____

Second Panel Member
Panel Member
Date signed: _____

Third Panel Member
Panel Member
Date signed: _____

Adrian Leo Pajarillo
Project Advisor
Date signed: _____

The Senior Project entitled

Haskell versus Rust: Benchmarking Efficient Concurrent Pathfinding in a Purely-Functional Programming Language

developed by

Prince Bernie B. Colis
Bachelor of Science in Computer Science

John Kenneth S. Lesaba
Bachelor of Science in Computer Science

Jon Ariel N. Maravilla
Bachelor of Science in Computer Science

Karl Frederick R. Roldan
Bachelor of Science in Computer Science

and submitted in partial fulfillment of the requirements of their respective Bachelor of Science degrees is hereby approved and accepted by the Department of Computer Science, College of Computer Studies, Ateneo de Naga University.

Marianne P. Ang, MS
Chair, Department of Computer Science
Date signed: _____

Joshua C. Martinez, MIT
Dean, College of Computer Studies
Date signed: _____

Declaration of Original Work

We declare that the Senior Project entitled

Haskell versus Rust: Benchmarking Efficient Concurrent Pathfinding in a Purely-Functional Programming Language

which we submitted to the faculty of the

Department of Computer Science, Ateneo de Naga University

is our own work. To the best of our knowledge, it does not contain materials published or written by another person, except where due citation and acknowledgement is made in our senior project documentation. The contributions of other people whom we worked with to complete this senior project are explicitly cited and acknowledged in our senior project documentation.

We also declare that the intellectual content of this senior project is the product of our own work. We conceptualized, designed, encoded, and debugged the source code of the core programs in our senior project. The source code of third party APIs and library functions used in my program are explicitly cited and acknowledged in our senior project documentation. Also duly acknowledged are the assistance of others in minor details of editing and reproduction of the documentation.

In our honor, we declare that we did not pass off as our own the work done by another person. We are the only persons who encoded the source code of our software. We understand that we may get a failing mark if the source code of our program is in fact the work of another person.

Prince Bernie B. Colis

3 - Bachelor of Science in Computer Science

John Kenneth S. Lesaba

3 - Bachelor of Science in Computer Science

Jon Ariel N. Maravilla

3 - Bachelor of Science in Computer Science

Karl Frederick R. Roldan

3 - Bachelor of Science in Computer Science

This declaration is witnessed by:

Adrian Leo Pajarillo

Project Advisor

Haskell versus Rust: Benchmarking Efficient Concurrent Pathfinding in a Purely-Functional Programming Language

by

Prince Bernie B. Colis, John Kenneth S. Lesaba, Jon Ariel N. Maravilla, and Karl Frederick R. Roldan

Project Advisor: Adrian Leo Pajarillo

Department of Computer Science

EXECUTIVE SUMMARY

To be filled in later. `/*TODO*/`.

I dedicate this research work to all of humanity.

ACKNOWLEDGEMENTS

I thank everyone who helped me finish this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Since 2010, functional programming has been getting more popular (Purescript, Typescript, ReactJS, Elm), people are trying to find more ways of leveraging the relatively easier-reasonability of functional programming without having to sacrifice program performance. Directly proportional to the trend is the developing advancements in time and space complexity of different programming languages and those events are inevitable due to the fact that we live in a technological age. This paper's goal is to know how can a lazy-first purely functional programming language have comparable performance with respect to an eager memory-safe language in terms of multi-threaded programming? In turn helps in the said advancements in different programming languages.

1.1 Project Context

Reasoning about program correctness in a pure function can be done in a dependently-typed, proof assistant such as Coq or Agda[2][12][4]. Likewise, pure functions can be easily proved by using induction. Composing two proven functions into a single function should also give the correct result[1].

As more people use functional programming languages, the need for pure function algorithms has greatly increased. Most references such as *Introduction to Algorithms*, *The Algorithm Design Manual*, and *The Art of Computer Programming* present algorithms in a mostly imperative or structural manner.[3][11][8] Thus, most programmers are more familiar with imperative approaches to programming.

1.2 Purpose and Description

This research aims to utilize the existing A* pathfinding algorithm [5][13] and find a way to develop a reasonably-efficient purely-functional implementation of the algorithm using parallel data structures such as STMs or MVars[9].

The A* Pathfinding algorithm is used heavily in video games, telephone traffic, and other graph traversal problems[6]. This research aims to aid in the development of video games using the functional programming paradigm in the future as video game development is dominated by imperative languages.

1.3 Objectives

The main objective of the research is to find an efficient concurrent implementation of a maze solver in a purely-functional programming environment with comparable performance and space complexity with respect to an efficient imperative programming language. The researchers also aim to compare performances between Haskell and Rust by creating a program such as a maze-solver that can be benchmarked using both programming languages, then gather necessary information and come up with a detailed analysis regarding the performances of both languages where the number of CPU cores and threads as independent variables.

1.4 Scope and Limitations

The researchers will utilize Haskell for concrete implementation of the parallel A* algorithm in a functional programming environment. Likewise, for performance comparison, the researchers will use the Rust programming language due to some of its features having similarities with Haskell such as correct concurrent programs[10] and guarantees a relatively safe program[7].

Other purely-functional languages or lambda notation for generality will not be used. Other concurrent data structures besides `MVar` and Software Transactional Memory will not be utilized. Likewise, only mazes with a reachable end-goal will be tested since if the A* algorithm runs on unbounded mazes, that is - having no reachable end goal, the algorithm will not halt.[6]

The concrete implementation and analysis is planned to be tested on only one CPU, namely Intel Core i7-9750H CPU at 2.60Ghz with 6 cores and 12 logical threads.

Appendix A

Code Listing

REFERENCES

[1] A. Abel, M. Benke, A. Bove, J. Hughes, and U. Norell, *Verifying haskell programs using constructive type theory*, 01 2005, pp. 62–73.

[2] J. Breitner, A. Spector-Zabusky, Y. Li, C. Rizkallah, J. Wiegley, and S. Weirich, *Ready, set, verify! applying hs-to-coq to real-world haskell code*, 2018.

[3] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*, The MIT Press, 3rd ed., 2009.

[4] Y. El Bakouny, T. Crolard, and D. Mezher, *A coq-based synthesis of scala programs which are correct-by-construction*, Proceedings of the 19th Workshop on Formal Techniques for Java-like Programs, (2017).

[5] Z. et al., *Parallelizing a* path finding algorithm*, International Journal Of Engineering And Computer Science, 6 (2017), pp. 22469–22476.

[6] P. E. Hart, N. J. Nillson, and R. Betram, *A formal basis for the heuristic determination of minimum cost paths*, IEEE Transactions of Systems Science and Cybernetics, SSC-4 (1968).

[7] R. Jung, J.-H. Jourdan, R. Krebbers, and D. Dreyer, *Rustbelt: Securing the foundations of the rust programming language*, Proc. ACM Program. Lang., 2 (2017).

[8] D. E. Knuth, *The Art of Computer Programming, Volume 1 (3rd Ed.): Fundamental Algorithms*, Addison Wesley Longman Publishing Co., Inc., USA, 1997.

[9] S. Marlow, *Parallel and Concurrent Programming in Haskell*, O'Reilly Media, Inc., 2013.

[10] A. Saligrama, A. Shen, and J. Gjengset, *A practical analysis of rust's concurrency story*, 2019.

[11] S. S. Skiena, *The Algorithm Design Manual*, Springer, London, 2008.

[12] A. Spector-Zabusky, J. Breitner, C. Rizkallah, and S. Weirich, *Total haskell is reasonable coq*, Proceedings of the 7th ACM SIGPLAN International Conference on Certified Programs and Proofs, (2018).

[13] A. Weinstock and R. Holladay, *Parallel a* graph search.*

VITA

`/*TODO*/` are BS Computer Science student of the Department of Computer Science at the Ateneo de Naga University.