

Exploring Parallel MCTS on Chess Game

曾正豪 0716325
CS NYCU
Hsinchu, Taiwan

王健業 0716098
CS NYCU
Hsinchu, Taiwan

張宸愷 0710018
EECSHP NYCU
Hsinchu, Taiwan

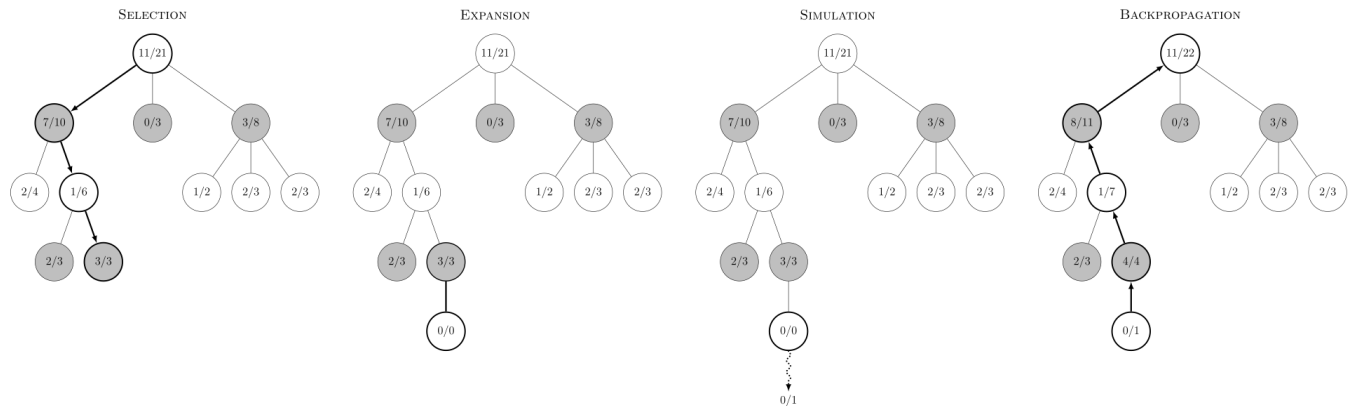


Figure 1: Illustration of a single step of MCTS

ABSTRACT

A project proposal for the course 'Parallel Programming Fall 2021'. We decided to explore the parallelization of MCTS using the techniques and knowledge we have learned in this course. We will use quantitative benchmarks to compare different approaches to solve this kind of parallelization.

KEYWORDS

MCTS, parallel programming, Pthreads, CUDA

ACM Reference Format:

曾正豪 0716325, 王健業 0716098, and 張宸愷 0710018. 2021. Exploring Parallel MCTS on Chess Game. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

One of the most famous applications of MCTS is AlphaGo. It uses Monte Carlo tree search with 2 other neural networks to play Go. An early version of AlphaGo was tested on hardware with various numbers of CPUs and GPUs, running in asynchronous or distributed mode. It was tested with search threads from 12 to 64, number of CPUs from 48 to 1920, and number of GPUs from 1 to 280. And in 2016, it changed to use TPUs (tensor processing units)

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, July 2017, Washington, DC, USA
© 2021 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00
<https://doi.org/10.1145/1122445.1122456>

as its computing unit. In recent years, it keeps beating many go players. Overall, MCTS is an algorithm that can be highly parallelized. Hence, we decided to use MCTS as the topic of our final project.

2 STATEMENT OF PROBLEM

3 PROPOSED APPROACHES

4 LANGUAGE SELECTION

5 RELATED WORK

A pretty detailed work [1]. Different methods for enhancing the capability of MCTS on board games were explored in [2]. It also outlined some of the parallelization techniques for MCTS.

6 EXPECTED RESULTS

7 TIMETABLE

8 ACKNOWLEDGMENTS

Identification of funding sources and other support, and thanks to individuals and groups that assisted in the research and the preparation of the work should be included in an acknowledgment section, which is placed just before the reference section in your document.

9 APPENDICES

REFERENCES

- [1] Anji Liu, Yitao Liang, Ji Liu, Guy Van den Broeck, and Jianshu Chen. 2020. On Effective Parallelization of Monte Carlo Tree Search. [arXiv:2006.08785](https://arxiv.org/abs/2006.08785) [cs.LG]
- [2] Martin Weigel. 2017. Monte Carlo methods for massively parallel computers. [arXiv:1709.04394](https://arxiv.org/abs/1709.04394) [physics.comp-ph]

A ONLINE RESOURCES

- (1) Monte Carlo tree search (https://en.wikipedia.org/wiki/Monte_Carlo_tree_search)